Aus dem Institut für Medizinische Informationsverarbeitung, Biometrie und Epidemiologie der Ludwig–Maximilians–Universität München

Vorstand: Prof. Dr. rer. nat. Ulrich Mansmann

Einsatz und Optimierung einer überwachten Klassifizierungsmethode im Kontext eines Privacy-Preserving-Record-Linkage

Dissertation
zum Erwerb des Doktorgrades der Humanbiologie
an der Medizinischen Fakultät der Ludwig-Maximilians-Universität zu München

vorgelegt von
Daniel Nasseh

aus
München
2014
Mit Genehmigung der Medizinischen Fakultät
der Universität München

Berichterstatter: Prof. Dr. Jürgen Stausberg

Mitberichterstatter: Priv. Doz. Dr. Klaus Adelhard
Priv. Doz. Dr. Stefan Wirth

Dekan: Prof. Dr. med. Dr. h.c. M. Reiser FACR FRCR

Tag der mündlichen Prüfung: 26.11.2014
Publikationen im Umfeld dieser Arbeit


Inhaltsverzeichnis

PUBLIKATIONEN IM UMFELD DIESER ARBEIT ............................................................... 3

INHALTSVERZEICHNIS .................................................................................................. 4

1. EINLEITUNG .............................................................................................................. 6
   1.1. EINFÜHRUNG IN DIE THEMATIK ........................................................................ 6
   1.2. MOTIVATION ZUR DURCHFÜHRUNG DER VORLIEGENDE ARBEIT ......................... 8
       1.2.1. Studie zu familiärem Darmkrebs ........................................................................ 8
       1.2.2. Klassifizierungsproblematik während der DKFS .............................................. 11
   1.3. GRUNDLAGEN DES PRIVACY-PRESERVING-RECORD-LINKAGE ......................... 16
       1.3.1. Historischer Hintergrund .................................................................................. 16
       1.3.2. Technischer Ablauf des Privacy-Preserving-Record-Linkage .............................. 17
       1.3.3. Klassifikationstechniken .................................................................................. 28
       1.3.4. Softwaresysteme im Bereich des Data-Matchings ............................................. 31
       1.3.5. Möglichkeiten der Evaluation .......................................................................... 32
   1.4. ZIELSETZUNG ...................................................................................................... 34

2. MATERIAL UND METHODEN ................................................................................... 36
   2.1. VOREBEREITENDE ARBEITEN UND ARBEITSMATERIAL ..................................... 36
       2.1.1. Verwaltung der Arbeitsumgebung ..................................................................... 36
       2.1.2. Record-Linkage: Spezifikation und Implementierung ....................................... 36
       2.1.3. Beschreibung der verwendeten klinischen Daten .............................................. 39
   2.2. ÜBERWACHTE KLASSEFIZIERUNG – ANGESTREBTES VORGEHEN ...................... 40
   2.3. ERZEUGUNG VON TESTSETS ANHAND KLINISCHER DATEN .................................. 42
       2.3.1. Notwendigkeit der Testset-Erzeugung ................................................................. 42
       2.3.2. Spezifizierung der Parameter zur Testset-Erzeugung ......................................... 43
       2.3.3. Konkrete Implementierung der Testset-Erzeugung ............................................ 46
       2.3.4. Auswertung der Testsets .................................................................................. 51
   2.4. IDENTIFIKATION VON POTENTIEL EINFLUSSREICHEN PARAMETERN AUF DIE ERZEUGUNG VON TRAININGSSETS .................. 53
   2.5. ÜBERPRÜFUNG DES EINFLUSES VON KONSTRUKTIONSPARAMETERN AUF DIE QUALITÄT DER KLASSEFIZIKATION ......................................................... 55
       2.5.1. Zielsetzung der Parameterprüfung ...................................................................... 55
       2.5.2. Erstellen von Template-Trainingssets ............................................................... 57
       2.5.3. Variation der Größe .......................................................................................... 60
       2.5.4. Variation der Fehlerrate ................................................................................... 60
       2.5.5. Variation der Überlappung ............................................................................... 60
       2.5.6. Variation der Verteilung .................................................................................. 61
       2.5.7. Performanzvergleich der Klassifikatoren der Trainingsset-Varianten .................. 62
   2.6. VERGLEICH VON UNÜBERWACHTER KLASSEFIZIERUNG MIT ANDEREN KLASSEFIZIKATIONSTECHNIKEN ................................................. 62
       2.6.1. Zielsetzung des Klassifikatorenbgleichs ............................................................. 62
       2.6.2. Überwachte Klassifizierung der Testdaten ......................................................... 63
<table>
<thead>
<tr>
<th>2.6.3. Unüberwachte Klassifizierung der Testdaten</th>
<th>63</th>
</tr>
</thead>
<tbody>
<tr>
<td>3. ERGEBNISSE</td>
<td>68</td>
</tr>
<tr>
<td>3.1. TESTSET-ERZEUGUNG</td>
<td>68</td>
</tr>
<tr>
<td>3.2. AUF TRAININGSSET-VARIANTEN BASIERENDE KLASSEIFIKATIONSERGEBNISSE</td>
<td>72</td>
</tr>
<tr>
<td>3.3. CLARA</td>
<td>77</td>
</tr>
<tr>
<td>3.4. VERGLEICH VERSCHIEDENER KLASSEIFIKATIONSMETHODEN</td>
<td>79</td>
</tr>
<tr>
<td>4. DISKUSSION</td>
<td>83</td>
</tr>
<tr>
<td>4.1. BEGRÜNDUNG DER KONZEPTION EINES ÜBERWACHTEN KLASSEIFIKATIONSSYSTEMS</td>
<td>83</td>
</tr>
<tr>
<td>4.2. ZUGRUNDELIEGENDE ARBEITSMATERIALIEN</td>
<td>84</td>
</tr>
<tr>
<td>4.3. HYPOTHESE ALS AUSGANGSPUNKT DES WISSENSCHAFTLICHEN VORGANGS</td>
<td>86</td>
</tr>
<tr>
<td>4.4. ABGLEICH UND BEWERTUNG VERSCHIEDENER KLASSEIFIKATOREN</td>
<td>88</td>
</tr>
<tr>
<td>4.5. ÜBERTRAGUNG DER ERGEBNISSE AUF DEN AKTUELLEN STAND DER WISSENSCHAFT</td>
<td>90</td>
</tr>
<tr>
<td>4.6. LIMITIERUNGEN DER ARBEIT</td>
<td>91</td>
</tr>
<tr>
<td>5. ZUSAMMENFASSUNG</td>
<td>93</td>
</tr>
<tr>
<td>6. LITERATURVERZEICHNIS</td>
<td>94</td>
</tr>
<tr>
<td>7. ANHANG</td>
<td>100</td>
</tr>
<tr>
<td>DANKSAGUNG</td>
<td>114</td>
</tr>
<tr>
<td>EIDESSTATTLICHE VERSICHERUNG</td>
<td>115</td>
</tr>
</tbody>
</table>
1. Einleitung

1.1. Einführung in die Thematik

Das Erzeugen, Sammeln und Weitergeben von Daten in großem Stil ist heute selbstverständlicher Bestandteil unseres alltäglichen Lebens geworden. Man denke nur etwa an die vielen bereits in die Milliarden [1] gehenden Online-Profile auf Facebook oder anderen sozialen Netzwerken, auf denen persönliche Daten freiwillig geteilt und veröffentlicht werden [2].


Gerade in der Medizin kommt dem Datenschutz eine immens hohe Bedeutung zu, da es sich bei medizinischen Daten um Daten mit sensiblen Inhalt (§ 3 Abs. 9 BDSG) handelt. Als sensible Daten bezeichnet man generell Daten mit Angaben über die rassische und ethnische Herkunft, politische Meinung, religiöse oder philosophische Überzeugung, Gewerkschaftszugehörigkeit, Gesundheit oder Sexualleben.


Es stellt sich nun die Frage, wie es im Zuge von medizinischer Forschung ermöglicht werden kann, auf Patientendaten, die einem Schutzversprechen unterliegen, unter Beachtung desselben zuzugreifen. Kohorten-Studien, wie sie beispielsweise im Zuge des KORA- Projektes oder der deutschen Kohorte stattfinden [7,8], arbeiten direkt mit Probanden, die ihre Daten unter Erklärung ihres Einverständnisses zur Verfügung stellen. Die Daten werden hierzu in
Einleitung

vorbereiteten Studienzentren erfasst. Ein Datenzugriff ist also zu Forschungszwecken grundsätzlich möglich.


Eine weitere große Herausforderung zeigt sich, wenn medizinische Daten bereits existieren und mit medizinischen Daten aus anderen Datenquellen zusammengeführt werden sollen um etwa mögliche Zusammenhänge zwischen den Daten zu erkennen. Solche Szenarien treten zum Beispiel dann auf, wenn Studiendaten zusätzlich mit Registerdaten verknüpft werden

Als konkretes Beispiel für die Notwendigkeit eines solchen Record-Linkage-Verfahrens stellte sich dem Verfasser dieser Arbeit eine Studie zu familiärem Darmkrebs in München dar (siehe Kapitel 1.2.1) [21]. Während der Mitarbeit an der genannten Studie eröffneten sich im Bereich des Record-Linkage einige wissenschaftlich interessante Fragestellungen. Vor allem bezüglich der Klassifizierung, einem wesentlichen Teilbereich des Record-Linkage-Prozesses, konnte Verbesserungspotential bezüglich des Standes der Wissenschaft identifiziert werden, was zu einer Reihe von weiterführenden Untersuchungen, Analysen und Entwicklungen bezüglich der Klassifizierung im Bereich des Privacy-Preserving-Record-Linkage motivierte.

1.2. Motivation zur Durchführung der vorliegenden Arbeit

1.2.1. Studie zu familiärem Darmkrebs

Medizinischer Hintergrund

es sich hingegen um einen weiteren Risikofaktor, der unabhängig von bekannten genetischen Dispositionen dazu führt, dass diese Erkrankung in Familien oftmals gehäuft auftritt [26].


**Zielsetzung und grober Ablauf der Studie**


Leider lassen sich die Familienbeziehungen innerhalb des TRMs nicht rekonstruieren, da notwendige Daten zur Familienstruktur nicht im Register abgespeichert werden. Es gilt also, die im TRM hinterlegten medizinischen Daten (MDAT) der Angehörigen und Indexpatienten mit den Studiendaten, unter Erhalt der Familienstruktur, über andere Wege in Beziehung zu setzen.

Mittels spezieller Erfassungsbögen (siehe Abbildung 1) werden die identifizierenden Daten (IDAT) naher Verwandter der neu erkrankten, an der Studie teilnehmenden Indexpatienten im Einzugsgebiet des TRM erfasst.

*Datenschutzkonzept der Studie*


Abbildung 1: Datenerfassungsbogen der DKFS.

1.2.2. Klassifizierungsproblematik während der DKFS

In der DKFS gab es eine Reihe von Aspekten, die im Bezug auf das Privacy-Preserving-Record-Linkage Probleme bereiteten. Ein Hauptproblem zeigte sich bei der Festlegung einer binären Schranke, die die Menge der potentiellen Links in echte bzw. falsche Links unterteilt. Die Festlegung einer binären Schranke ist Teil des Klassifizierungsprozesses des Privacy-Preserving-Record-Linkage, wobei der jetzige Stand der Wissenschaft keine eindeutige, standardisierte Lösung für dieses Problem präsentieren kann [38]. Das Klassifizierungsproblem wird nachfolgend im Bezug auf die Familienstudie eingehend erläutert. Kapitel 1.3.3 beschäftigt sich zudem mit dem generellen Stand der Wissenschaft zum Klassifizierungsprozess im Bereich des Record-Linkage, insbesondere mit Augenmerk auf binäre Klassifikation (Unterteilung aller Links in zwei Klassen – echte Links und falsche Links).

Abbildung 3: Darstellung verschiedener möglicher Histogramme zur Erläuterung der während des Rekord-Linkage auftretenden Klassifikationsproblematik.
Einleitung


Im Falle der DKFS war die Klassifikation besonders problembehaftet, da die Daten der Angehörigen der Patienten nicht direkt von den Angehörigen, sondern stellvertretend durch die Indexpatienten über Aufnahmebögen (siehe Abbildung 1) oder telefonisch gesammelt wurden. Oftmals fehlten den Patienten hierbei die exakten Informationen, wie beispielsweise der genaue Wohnort, oder das exakte Geburtsdatum ihrer Angehörigen, es wurden jedoch trotzdem Angaben gemacht, die dem nachfolgenden Record-Linkage jedoch eher abträglich waren. Die während des Klassifikationsprozesses erstellten Histogramme während des Record-Linkage zwischen Studien- und Registerdaten entsprachen also nicht dem Optimalbeispiel aus Abbildung 3a, sondern eher den Problemfällen wie sie unter Abbildung 3b bzw. Abbildung 3c wiedergegeben wurden. Abbildung 4 zeigt diesbezüglich eines der Histogramme der Menge aller Links zum Record-Linkage-Durchlauf am 04.02.2014. Es ist hierbei anzumerken, dass für die Klassifikation innerhalb des DKFS Projektes insgesamt 9 verschiedene Histogramme verwendet werden, die unter anderem eine differenzierte Ansicht von Angehörigen und Patienten erlauben.

Für solch einen Vergleich sind jedoch Klartextdaten notwendig, welche im Kontext des probabilistischen Privacy-Preserving-Record-Linkage, also auch in Bezug auf die DKFS, nicht gegeben waren. Anhand der hier vorkommenden, einwegverschlüsselten Daten ließ sich lediglich beurteilen, ob Attribute vollkommen übereinstimmen oder nicht. Im Falle der DKFS wurde die Information der einzelnen Attributübereinstimmungen im Unsicherheitsbereich (jedoch ohne Klartextinformation) unterstützend bei der Schrankenfindung mitverwendet (siehe Abbildung 5). Die Datei beinhaltete detaillierte Angaben zu linkspezifischen Übereinstimmungen (J), Nicht-Übereinstimmungen (N) und fehlenden Werten auf Seiten der Studiendaten bzw. des TRM (SF=Studie fehlt, TF=TRM Daten fehlen, BF=Daten fehlen auf beiden Seiten). Werte in Klammern standen für die Häufigkeit der jeweils genannten Angaben in Attributen in denen Mehrfachvorkommen möglich sind. Nach Durchsicht der Histogramme wurde die Datei genutzt um die Bestimmung des exakten Punktes des binären Klassifikators zu unterstützen. Im gegebenen Beispiel wurde die Schranke auf 24.9 festgelegt. Der Ausschnitt ist weder in der Zahl der Einträge noch in der Menge der Spalten vollständig.

Tabelle 1: Unterschiedliche Darstellung einer Entität in zwei verschiedenen Datenbanken.

<table>
<thead>
<tr>
<th></th>
<th>Datenset 1</th>
<th>Datenset 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nachname</td>
<td>SMITH</td>
<td>SMYTH</td>
</tr>
<tr>
<td>Vorname</td>
<td>ALAN</td>
<td>ALAN</td>
</tr>
<tr>
<td>Geschlecht</td>
<td>M</td>
<td>M</td>
</tr>
</tbody>
</table>
Abbildung 4: Eines der konkreten Histogramme zum Record-Linkage der DKFS am 04.02.2014.

Auch wenn für die DKFS bei der manuellen Schrankensetzung und somit bei einem gewissen Maß an Unsicherheit vorerst verblieben wurde, wäre es wünschenswert, automatisierte, binäre Klassifikationsvarianten entscheidungsunterstützend in den Klassifikationsprozess einzubringen.

Leider existieren keine vergleichenden Analysen zu diesen Methoden, und es ist unklar, ob die Methoden überhaupt zur manuellen Klassifikation verbessernd beitragen können. Der Stand der Wissenschaft zu genannten Klassifikationsmethoden wird weiterführend unter Kapitel 1.3.3 beschrieben.

1.3. Grundlagen des Privacy-Preserving-Record-Linkage

1.3.1. Historischer Hintergrund


Variationen in Attributwerten, sowie Möglichkeiten der Abschätzung von Fehlerhäufigkeiten mittels automatisierter Methoden präsentierte.


1.3.2. Technischer Ablauf des Privacy-Preserving-Record-Linkage

Technisch werden beim Record-Linkage in der Regel Einträge zweier Datensets zueinander zugeordnet. Der Ablauf des Record-Linkage lässt sich in vier rudimentäre Arbeitsschritte einteilen:

- Vorverarbeitung
- Blocking/Indexing
- Gewichtsbestimmung
- Klassifikation
Einleitung

Der eben genannte technische Ablauf wird grafisch in Abbildung 6 grob wiedergegeben. Die einzelnen Arbeitsschritte werden in den nachfolgenden Unterkapiteln weiterführend beschrieben.

Standardisierung (Vorverarbeitung I)


Bei Personen identifizierenden Daten im medizinischen Sektor werden die ursprünglichen Datenfelder nach bestimmten Regeln standardisiert. Der UNICON-Regelsatz [55] wäre hierbei z.B. der Regelsatz, der in der DKFS Studie inklusive einiger szenarienspezifischer Anpassungen
verwendet wurde. Hierbei sind folgende Anweisungen zu nennen, die während der Standardisierung umgesetzt werden.

- Ersetzung undeutscher Sonderzeichen (basierend auf ausgewählten Listen) in das deutsche Äquivalent (Bsp.: é -> e).
- Uniforme Großschreibung (Bsp.: Hans-Wagner -> HANS-WAGNER).
- Umlaut-Normalisierung (Bsp.: FÖRSTER -> FOERSTER).
- Ersetzung von Trennsymbolen durch Leerzeichen (Bsp.: HANS-WAGNER -> HANS WAGNER).
- Erkennung spezifischer Schlagwörter. Dieser Schritt ist feldspezifisch. Im Feld „Titel“ werden hierbei beispielsweise nur gültige Titel (basierend auf einer zuvor erstellten Liste) zur weiteren Verarbeitung zugelassen. (Bsp.: Dr.)
- Konsistenz- bzw. Formatprüfung. (Bsp.: Entfernung des Geburtsdatums bei 33.02.19083)

Weiterhin ist es möglich, nach phonetischen Kriterien zu standardisieren. Somit werden Namensvarianten wie beispielsweise „Meyer“, bzw. „Meier“, die phonetisch übereinstimmen, in eine standardisierte Variante umgewandelt. Algorithmen, die hierzu verwendet werden, sind im englischsprachigen Raum der SOUNDEX [56] bzw. im deutschsprachigen Raum die Kölner Phonetik [57].

**Einwegverschlüsselung (Vorverarbeitung II)**

Einleitung


Tabellen 2 illustriert die Ausgabe zu verschiedenen Eingabewerten in Hexadezimalschreibweise, basierend auf der SHA-256 Funktion. Trotz der hohen Textähnlichkeit der Ausgangswerte im vorliegenden Beispiel erzeugt die Hash-Funktion komplett unterschiedliche Rückgabewerte.

Tabelle 2: Anwendung des SHA-256 auf verschiedene Ausgangswerte.

<table>
<thead>
<tr>
<th>Ausgangswert</th>
<th>Hash-Wert</th>
</tr>
</thead>
<tbody>
<tr>
<td>Meier</td>
<td>05c2d2b4cad1a3f5bf547b484ac6f4a70893e944d5bd6fe0f28db40453bf3f3c</td>
</tr>
<tr>
<td>Meyer</td>
<td>876dfa1d1152c1d0243866a1f66e7725f292ef83404fc4d3be79c1b51cc81c45</td>
</tr>
</tbody>
</table>


Das approximative Record-Linkage, das eine Weiterentwicklung des probabilistischen Privacy-Preserving-Record-Linkage darstellt, ersetzt die Einwegverschlüsselung basierend auf Hash-Werten durch Bloom-Filter [14,52]. Bloom-Filter sind Bit-Arrays, also Speicherstrukturen mit einer festgelegten Länge und einer Indexstruktur. Die Feldwerte des Arrays lassen sich dabei mit Bit-Werten, also mit 0 oder 1, belegen.

Initialisiert werden die Bloom-Filter in jedem Feld mit einem 0-Wert. Die Technik basiert darauf, die zu verschlüsselnden Wortketten in Q-gramme (in der Regel Bi-gramme) zu zerlegen.

Die Berechnung der Gewichte sowohl beim Kontrollnummer- als auch auf Bloom-Filter-Abgleich wird im nachfolgenden Unterkapitel zur Gewichtsberechnung weiter diskutiert.

Abbildung 7: Einwegverschlüsselung von Werteausprägungen anhand von Bloom-Filtern.
**Blocking/Indexing**

Die Zuordnung von Einträgen innerhalb zweier Datensets A und B erfolgt im Grunde genommen durch den Abgleich jeweils eines Eintrages aus A mit allen Einträgen aus B. Die Menge an notwendigen Vergleichen ist also das Kreuzprodukt der Anzahl an Einträgen aus A und B:

\[ |A| \times |B| \]


Beim Canopy-Clustering [70] werden Werte, die sich in der Blocking-Variable ähneln, in denselben Cluster eingefügt und innerhalb dieses Clusters abgeglichen. Dieses Verfahren ist allerdings nicht auf einwegverschlüsselte Daten übertragbar, da die verwendeten Ähnlichkeitsmaße Klartextdaten voraussetzen.

Als Nebeneffekt hat das Blocking auch Einfluss auf Qualitätswerte, vor allem auf die Anzahl der True-Negatives, die zur Evaluation des Record-Linkage verwendet werden können (siehe Kapitel 1.3.4). Da bei Anwendung von gut gewählten Blocking-Varibalen die Anzahl der True-Positives, False-Positives sowie False-Negatives meist nur leicht variiert, sich aber in der Anzahl der True-Negatives gewaltig reduziert, ist vor allem die Spezifität hiervon betroffen. Da die Spezifität beim Record-Linkage meist jedoch nahe der 100% liegt, verwendet man aber generell lieber den F-Measure-Wert, der unabhängig von der Spezifität, bzw. von den True-Negatives fungiert [71].

### Gewichtsbestimmung

Im Gegensatz zum deterministischen Record-Linkage stellt sich das probabilistische Record-Linkage als fehlertoleranter dar. Hierbei wird für jeden paarweisen Abgleich der Kontrollnummern zwischen den zu vergleichenden Einträgen ein Einzelgewicht berechnet und anhand der Summe dieser Einzelgewichte wird der Eintrag als echter bzw. falscher Link klassifiziert (siehe Formel 1).

\[ w = \sum w_i \]  

(1)

Für die Erläuterung der Berechnung der Einzelgewichte sind einige initiale Definitionen notwendig. Während A und B die zu vergleichenden Datensets repräsentieren, stehen die Mengen M und U für die Menge der Übereinstimmungen bzw. der Nicht-Übereinstimmungen (siehe Formel 2-4).

\[ A \times B = \{(a, b); a \in A, b \in B\} \]  

(2)

\[ M = \{(a, b); a = b, a \in A, b \in B\} \]  

(3)

\[ U = \{(a, b); a \neq b, a \in A, b \in B\} \]  

(4)

Bei \(a_1, \ldots, a_n\) bzw. \(b_1, \ldots, b_n\) handelt es sich um die einzelnen Attribute zu den Einträgen a bzw. b, aus Datenset A bzw. B (siehe Formel 5).

\[ a = (a_1, \ldots, a_n), b = (b_1, \ldots, b_n) \]  

(5)


\[ u_{ik} = P(a_i = b_i \land a_i = x_{ik} | (a, b) \in U) \]  

(6)

Der u-Wert beschreibt konkret die Wahrscheinlichkeit, dass zwei Einträge im Merkmal i mit der Ausprägung \(x_{ik}\) übereinstimmen und es sich dabei nicht um dieselbe Person/Eintrag handelt. Die u-Werte lassen sich hierbei im praktischen Umgang direkt aus der Häufigkeit von
Einleitung


Die in den Ausprägungen auftretenden Fehlerhäufigkeiten, die ebenfalls zur Gewichtsberechnung benötigt werden, lassen sich durch die m-Werte repräsentieren (siehe Formel 7).

\[
m_{ik} = P(a_i = b_i \land a_i = x_{ik} | (a, b) \in M)
\]

Der m-Wert beschreibt hierbei konkret die Wahrscheinlichkeit, dass zwei Einträge im Merkmal i mit der Ausprägung \(x_{ik}\) übereinstimmen und es sich dabei um die selbe Person/Eintrag handelt. Die m-Werte lassen sich vereinfacht jedoch auch als invertierte Fehlerhäufigkeiten im jeweiligen Attribut interpretieren. Typischerweise haben Adressangaben eine relativ hohe Fehlerhäufigkeit. Würde man also zum Beispiel in einem Datensatz zu 10% der Fälle Fehler in den Adressangaben erwarten, wäre der hierzu gehörende m-Wert 0.9. Die m-Werte können entweder aus ähnlichen [39,73], bereits ausgewerteten Datenbeständen mit bekannten Fehlerhäufigkeiten oder mittels einer Variante des Expectation-Maximation-Algorithmus [74] abgeschätzt werden.

Anhand der u-Werte und m-Werte lassen sich schließlich die Einzelgewichte berechnen (siehe Formel 8 bzw. Formel 9).

\[
w_i = \log \left( \frac{m_i}{u_{ik}} \right), \text{falls } a_i = b_i \land a_i = x_{ik}
\]

\[
w_i = \log \left( \frac{1 - m_i}{1 - u_{ik}} \right), \text{falls } a_i \neq b_i \land a_i = x_{ik}
\]

Falls die vergleichenden Attributsausprägungen übereinstimmen, wird wie bereits erwähnt ein positives Gewicht berechnet, falls die vergleichenden Attributsausprägungen nicht übereinstimmen, wird ein negatives Gewicht berechnet. Zudem gilt: Stimmen Kontrollnummern in einer seltenen Ausprägungen überein, so resultiert dies in einem stärkeren Gewicht. Das Übereinstimmen in häufigen Ausprägungen kann eher auf Zufall basieren, demnach wird ein niedrigeres Gewicht vergeben. Je höher die abgeschätzte Fehlerrate in einem Attribut ist, umso unbedeutender, also niedriger ist das Gewicht im
Vergleich zu anderen Attributen mit geringeren Fehlerraten. Nach Aufaddieren der Einzelgewichte zu einem Gesamtgewicht kann schließlich klassifiziert werden.

Liegt das Gesamtgewicht eines Links über einem spezifischen Schrankenwert, so wird er als echter Link bewertet, unterhalb dieser Schwelle als falscher Link. Man spricht hierbei von einer binären Klassifikation (hierzu mehr unter Kapitel 1.3.2.). Das Auffinden dieses Schrankenwertes war eine nicht triviale Aufgabe und Hauptthematik dieser Arbeit. Im Gegensatz zum deterministischen Record-Linkage unterscheidet sich also das probabilistische Record-Linkage darin, dass es nicht in allen Kontrollnummern exakt übereinstimmen muss und somit zu einem gewissen Grad Fehler in den Daten zulässt. Die Spezifität erleidet hierbei in der Regel nur geringfügige Einbußen und liegt je nach Datensatz nahe 100%. Die Sensitivität kann durch das Tolerieren weniger Unstimmigkeiten im Vergleich zum deterministischen Record-Linkage enorm verbessert werden und liegt je nach Datensatz, nach einem systematischem Review von Silveira [75] bei den ausgewerteten Arbeiten zwischen 74-98%.

Dennoch besitzt das probabilistische Record-Linkage auf einwegverschlüsselten Daten Schwächen. Durch die Einwegverschlüsselung ist es grundsätzlich nicht möglich, die Ähnlichkeit zweier Ausprägungen zu gewichten. Da bereits kleine Fehler in den Werteausprägungen (z.B. Schmitt bzw. Schmidt) zu komplett unterschiedlichen Hash-Werten führen, ist es lediglich möglich, zu bewerten, ob die Werte übereinstimmen oder nicht (siehe hierzu auch Kapitel 1.2.2).

Das approximative Record-Linkage tritt dieser Problematik entgegen. Wie bereits unter Kapitel 1.3.2 beschrieben, werden die Ausprägungen alternativ zu den vorhergehenden Methoden mittels Bloom-Filtern einwegverschlüsselt. Der Abgleich erfolgt also nicht mehr wie bei den Vorgängervarianten auf Hash-Werten sondern auf den Bloom-Filtern. Dabei kann nicht nur wie auf Hash-Werten festgestellt werden, ob Werte generell übereinstimmen, sondern auch, wie sehr sich zwei Bloom-Filter ähneln.

Die Distanz zweier Bloom-Filter zueinander lässt sich mittels des Dice-Koeffizienten (siehe Formel 10), berechnen, der sich als passendes Distanzmaß bewiesen hat [14,76].

\[
D_{AB} = \frac{2h}{(a + b)}
\]  

(10)

Auf das Szenario des approximativen Record-Linkage übertragen entspricht h der Anzahl an Bitpositionen, die in beiden zu vergleichenden Bloom-Filtern (A,B) mit 1 belegt wurden, a ist die Anzahl an Bitpositionen, die ausschließlich in A mit 1 belegt wurden, wohingegen b die

**Binäre Klassifikation**

Im Falle des Privacy-Preserving-Record-Linkage ist durch die Einwegverschlüsselung der Ausgangsdaten oftmals eine manuelle Zuordnung unsicherer Links nicht möglich (siehe Kapitel 1.2.2). Dies resultiert in der Notwendigkeit von binärer Klassifikation, also im Normalfall in der Bestimmung eines spezifischen Schrankenwertes, der die Menge der Links, basierend auf ihrem Gewichtswert, in echte bzw. falsche Links einteilt. Die binäre Klassifikation ist jedoch nicht nur im Falle von unter Datenschutz befindlicher Daten notwendig sondern auch beim Einsatz vollautomatischer Systeme, bei denen keine manuelle Nachkontrolle möglich ist.

Die Rückgabe des probabilistischen bzw. approximativen Record-Linkage ist eine Liste von Links, bestehend aus einem Paar von Datenset spezifischen IDs, die eine Referenz auf den im jeweiligen Datenset beinhalteten Eintrag darstellen, sowie ein assoziiertes Gewicht, das
Aufschluss darüber gibt wie gut die beiden referentiellen Einträge zueinander passen (siehe Tabelle 3).

**Tabelle 3: Beispielhafte Darstellung des Inhaltes einer Gewichtsdatei.**

<table>
<thead>
<tr>
<th>ID A</th>
<th>ID B</th>
<th>Gewicht</th>
</tr>
</thead>
<tbody>
<tr>
<td>1252</td>
<td>5332</td>
<td>76,74</td>
</tr>
<tr>
<td>1773</td>
<td>6784</td>
<td>74,33</td>
</tr>
<tr>
<td>34</td>
<td>588</td>
<td>71,22</td>
</tr>
<tr>
<td>788</td>
<td>899</td>
<td>55,39</td>
</tr>
<tr>
<td>1899</td>
<td>1754</td>
<td>23,76</td>
</tr>
</tbody>
</table>


### 1.3.3. Klassifikationstechniken

**Unüberwachte Klassifikation**

für Klassengrenzen fehlinterpretiert werden können. Gerade einfache Methoden wie Clustering-Verfahren sind deswegen in Ihrer naiven Form eher ungeeignet.

Besser funktionieren sogenannte Active-Learning Ansätze [79], bei denen es sich formell um eine Hybridvariante aus unüberwachter und überwachter Klassifikation handelt, die aber im Grunde genommen eher den unüberwachten Methoden zuzuordnen wären. Hierbei werden sogenannte positive bzw. negative Keimmengen (Seeds) definiert. Diese enthalten Vergleiche, die zu einer hohen Wahrscheinlichkeit bzw. basierend auf szenariospezifisch definierten Kriterien ausschließlich echte bzw. falsche Übereinstimmungen darstellen. Diese Keimmengen werden dann als Trainingsdaten für die noch unklassifizierten Links verwendet, so dass diese basierend auf Algorithmen wie dem K-Nearest-Neighbour (KNN) oder Support-Vector-Maschinen (SVM) den Keimmengen zugeordnet werden können, bis alle Links schließlich klassifiziert wurden. Peter Christen konnte hierzu in einer Arbeit demonstrieren, dass diese Hybridansätze in der Lage sind, andere unüberwachte Techniken zu übertreffen [71].

Überwachte Klassifikation

Im Gegensatz zur unüberwachten Klassifizierung ist die überwachte Klassifizierung von den Gewichten der Originaldaten unabhängig und basiert auf im Vorfeld spezifizierten Trainingsdaten [41,80]. Hierzu werden Trainingssets benötigt, die in ihrer Beschaffenheit den zu klassifizierenden Daten ähneln und deren echte Übereinstimmungen durch das Teilen derselben ID in beiden Teilsets bekannt sind. Auf diesen Trainingssets lässt sich nun ein Record-Linkage durchführen und basierend auf ausgewählten Qualitätskriterien wie beispielsweise dem F-Measure eine optimale Schranke berechnen. Der Schrankenwert kann nun ebenfalls als Klassifikator für die Originaldaten verwendet werden. Alternativ ist es auch möglich, einen Entscheidungsbaum auf den Trainingsdaten zu generieren, anhand dessen Regeln erzeugt werden können, die die nachfolgende Klassifikation der Originaldaten ermöglichen [81].


Eine gute Ersatzmöglichkeit kann hierbei die künstliche Erzeugung von Trainingsdaten darstellen. Zum Erzeugen von Patientendaten gibt es sogar eigenständige Software-Kits, wie
z.B. die FEBRL-Toolbox, deren Personengenerierungsmodul auf aus Populationen entnommenen Verteilungswerten beruht [82]. Allerdings waren dem Autor keine Arbeiten bekannt, in denen ein solches Vorgehen, also überwachte Klassifikation auf künstlichen Trainingsdaten, in der Praxis tatsächlich umgesetzt wurde. Das Fehlen festgelegter Standards und der erhöhte Aufwand scheint viele Projektgruppen von überwachter Klassifikation zurückschrecken zu lassen.

An der Johannes-Gutenberg-Universität in Mainz finden Untersuchungen zu neuartigen überwachten bzw. semi-überwachten Klassifikationsmethoden statt [83,84]. Hierbei wird versucht, die Konzepte Bagging und Bumping auf das Szenario des Record-Linkage anzupassen. Bei Bagging und Bumping werden zu zufälligen Ziehungen aus Populationsverteilungen Klassifizierer generiert, deren Mittelwert als finaler Klassifizierer für die Originaldaten zu nutzen ist. Sariyar ist der Meinung, dass die überwachten Methoden dabei die unüberwachten Methoden übertreffen können, allerdings gibt es auch hier noch offene Fragen bezüglich der Parametrisierung, also der genauen Zusammenstellung dieser Trainingsdaten. So stellt zum Beispiel die genaue Festlegung der Anzahl der Trainingsdaten, die beim Bagging bzw. Bumping generiert werden, nach eigenen Angaben ein offenes Problem dar [84].

**Regelbasierte Klassifikation**


Hierdurch ist für die Methodik grundsätzlich keine Gewichtsdatei notwendig. Benötigt wird ausschließlich die Information, in welchen Attributen die Einträge übereinstimmen. Unterstützend wurde hierzu eine Variante in Form der Pair-Analysis-Datei in der DKFS verwendet (siehe Kapitel 1.2.2).

Bislang (Stand 2012) existiert noch keine ausgiebige vergleichende Prüfung der verschiedenen Klassifikationsmethoden auf verschiedenen Testsets [38].
1.3.4. **Softwaresysteme im Bereich des Data-Matchings**


*Tabelle 4: Übersicht frei zugänglicher Softwaresysteme im Bereich des Record-Linkage.*

<table>
<thead>
<tr>
<th>System</th>
<th>Beschreibung</th>
<th>Referenz</th>
</tr>
</thead>
<tbody>
<tr>
<td>Big Match</td>
<td>Dient dem Datenabgleich großer Datennengen. Besitzt jedoch kein User Interface.</td>
<td>[85]</td>
</tr>
<tr>
<td>D-Dupe</td>
<td>Ein graphisches Tool dessen Hauptaufgabe die Detektion von Duplikaten in Netzwerken und deren Subnetzwerken ist.</td>
<td>[86]</td>
</tr>
<tr>
<td>DuDe</td>
<td>Ein Toolkit bestehend aus mehreren Data-Matching Modulen. Dude besitzt kein grafisches Interface sondern ist als Erweiterung für Javaprojekte konzipiert.</td>
<td>[87]</td>
</tr>
<tr>
<td>FEBRL</td>
<td>Beinhaltet Algorithmen zur Datenvorverarbeitung, Deduplikation und dem Data-Matching. Der Fokus liegt hierbei auf der Anwendung für medizinische Datenbanken. Zudem ist es möglich mit FEBRL künstliche Testdaten anhand realer Verteilungswerte zu generieren.</td>
<td>[82]</td>
</tr>
<tr>
<td>FRIL</td>
<td>Stark parametrisierbare Data-Matching Software mit graphischem Interface. Teilweise schwierig in der Handhabung.</td>
<td>[88]</td>
</tr>
<tr>
<td>Mainzliste</td>
<td>Webbasierter Pseudonymisierungsdienst inklusive gewichtsbasiertem, modularem Record-Linkage System.</td>
<td>bitbucket.org/medinfo_mainz/mainzliste/</td>
</tr>
<tr>
<td>Merge ToolBox</td>
<td>Umfangreiches Data-Matching Paket, das die Anwendung von Privacy-Preserving-Record-Linkage mittels Bloom-Filtern gestattet. Die Module bauen teilweise auf der kommerziellen Software Stata auf.</td>
<td>[89]</td>
</tr>
</tbody>
</table>
### Systeme

<table>
<thead>
<tr>
<th>System</th>
<th>Beschreibung</th>
<th>Referenz</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OYSTER</strong></td>
<td>Wurde zur Erfassung und Verwaltung von Studentenakten erstellt. Enthält unter anderem Module für probabilistisches Record-Linkage.</td>
<td>[90]</td>
</tr>
<tr>
<td><strong>R RecordLinkage</strong></td>
<td>Paket für probabilistisches Record-Linkage für die Statistiksoftware „R“.</td>
<td>[91]</td>
</tr>
<tr>
<td><strong>SILK</strong></td>
<td>Umfangreiches Data-Matching System, das Daten im RDF Format speichert und abgleicht.</td>
<td>[92]</td>
</tr>
<tr>
<td><strong>Sim Metrics</strong></td>
<td>Beinhaltet eine große Auswahl approximativer Textvergleichsfunktionen.</td>
<td>sourceforge.net /projects/simmetrics</td>
</tr>
<tr>
<td><strong>TAILOR</strong></td>
<td>Umfangreiches Toolkit zu verschiedenen Anwendungen aus dem Bereich des Record-Linkage inklusive einiger Klassifikationsmethoden.</td>
<td>[93]</td>
</tr>
<tr>
<td><strong>WHIRL</strong></td>
<td>Beinhaltet einen regelbasierten Klassifikationsansatz.</td>
<td>[94]</td>
</tr>
</tbody>
</table>

### 1.3.5. Möglichkeiten der Evaluation

Das Hauptanliegen beim Datenabgleich ist das Erzielen einer möglichst hohen Abgleichs Qualität, durch die sich gleichzeitig die Güte von verschiedenen methodischen Ansätzen abschätzen und vergleichen lässt. Diese lässt sich anhand der Anzahl von echt bzw. falsch

<table>
<thead>
<tr>
<th>Klassifikation</th>
<th>Reality</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Übereinstimmung (MATCH)</td>
</tr>
<tr>
<td>Echter Link (LINK)</td>
<td>Echt Positive (TRUE POSITIVES)</td>
</tr>
<tr>
<td>Falscher Link (NON-LINK)</td>
<td>Falsch Negative (FALSE NEGATIVES)</td>
</tr>
</tbody>
</table>

**Abbildung 8: Kontingenztafel mit dem Urteil der Klassifikation und der tatsächlichen Klasse.**
ermittelten Übereinstimmungen, bzw. echt bzw. falsch ermittelten Nicht-Übereinstimmungen berechnen. Die vier beschriebenen Beobachtungen lassen sich übersichtlich in einer vier Felder Tafel, (*siehe Abbildung 8*) auf das Szenario des Record-Linkage angepasst, darstellen [33,95].

Durch die in der Vier-Felder Tafel aufgelisteten statistischen Maßeinheiten (True Positives (TP), False-Positives (FP), False-Negatives (FN), True-Negatives (TN)) lassen sich verschiedene Qualitätsmaße berechnen. Als häufig in der Statistik verwendete Qualitätsmaße wären hierzu die Spezifität sowie die Sensitivität zu nennen (siehe *Formel 11,12*):

\[
\begin{align*}
\text{Spezifität} &= \frac{TN}{TN + FP} \tag{11} \\
\text{Sensitivität} &= \frac{TP}{TP + FN} \tag{12}
\end{align*}
\]


Ein geeigneteres Qualitätsmaß im Kontext des Record-Linkage stellt deshalb der F-Measure-Wert da [71,96]. Hierbei handelt es sich um den harmonischen Mittelwert der Sensitivität und des positiv prädiktiven Wertes (siehe *Formel 13, 14*).
Einleitung

\[ PPV = \frac{TP}{TP + FP} \]  \hspace{1cm} (13)

\[ FM = \frac{2 \times PPV \times Sensitivität}{PPV + Sensitivität} \]  \hspace{1cm} (14)


1.4. Zielsetzung

Anhand einer Studie zu familiärem Darmkrebs (siehe Kapitel 1.2.1) wurden im Bereich des Record-Linkage Unsicherheiten bei der manuellen, binären Klassifikation, die zu einer Verminderung der Abgleichsqualität führen könnten, erkannt (siehe Kapitel 1.2.2). Unterstützend, oder auch alternativ, existieren bereits verschiedene automatisierte Klassifikationsansätze, nennenswert sowohl unüberwachte als auch überwachte Klassifikationssysteme (siehe Kapitel 1.3.3). Gerade zu überwachter Klassifikation existieren jedoch im Moment keine klaren Standards. Auch werden dort zusätzlich zu den Originaldaten Trainingsdaten benötigt.

Da reale Trainingsdaten meist nicht zur Verfügung stehen, könnten alternativ künstliche Trainingsdaten eingesetzt werden. Zu deren konkreter Beschaffenheit fanden sich jedoch keine Empfehlungen. Ausgangspunkt der Arbeit war die Überlegung, künstliche Trainingsdaten zu erzeugen, die den Originaldaten in hohem Maße ähneln. Basierend auf dieser Überlegung ergab sich die Zielsetzung, die optimale Parametrisierung bei der Konstruktion von künstlichen Trainingsdaten bei der überwachten Klassifizierung zu untersuchen und darauf aufbauend Empfehlungen zu erarbeiten.

Weiterhin fehlten Informationen und umfangreiche vergleichende Tests zur Performanz unüberwachter sowie überwachter Methoden im direkten Vergleich [38]. Das zu erarbeitende
überwachte Klassifikationssystem sollte deswegen mit verschiedenen, unüberwachten Klassifikationsansätzen sowie der manuellen Schrankengebung, wie sie in der DKFS Anwendung findet, verglichen werden.

Bei den zu vergleichenden unüberwachten Methoden sollte es sich sowohl um eine einfache Clustering-Methode, als auch um eine fortgeschrittene Technik aus dem Bereich des Active-Learnings, die anderen unüberwachten Methoden qualitativ überlegen ist, handeln [71].

Die Testdaten sollten sich in spezifizierten Parametern, der Größe, dem Überlappungsbereich, sowie der Fehlerhäufigkeit unterscheiden.
2. Material und Methoden

2.1. Vorbereitende Arbeiten und Arbeitsmaterial

2.1.1. Verwaltung der Arbeitsumgebung


Die Programme selber wurden kursiv und durch einen in spitzen Klammern nachfolgenden Index entsprechend Kapitel 7 – Anhang E im Text aufgeführt. Die Erstellung der in dieser Arbeit dargestellten Plots und einiger mathematischer Auswertungen erfolgte über die Statistik Software „R“ (http://www.r-project.org/).

Ein Abbild der finalen Arbeitsumgebung, also aller erzeugten Programme bzw. Klassen und Daten, wurde zur nachhaltigen Speicherung vom Autor dieser Arbeit gesichert und aufbewahrt. Für die teilweise zeitintensiven Berechnungen war ein leistungsstarker Rechner notwendig. Tabelle 5 skizziert die wichtigsten Hardwarekennziffern des zumeist verwendeten Systems.

Tabelle 5: Wichtigste Hardwarekomponenten des Arbeitssystems.

<table>
<thead>
<tr>
<th>Prozessor</th>
<th>Arbeitsspeicher</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intel(R) Core™ i7-3770 CPU @3,4 GHz</td>
<td>8 GB-RAM</td>
</tr>
</tbody>
</table>

2.1.2. Record-Linkage: Spezifikation und Implementierung

Für die zugrunde liegenden Tests und Entwicklungen wurde eine leicht abgewandelte Variante des probabilistischen Privacy-Preserving-Record-Linkage, das auch in der Familienstudie Anwendung fand, verwendet [67]. Hierbei handelte es sich um eine Implementierung des Fellegi und Sunther Algorithmus nach Spezifikation von Martin Meyer [31,39]. Die konkrete
Implementierung wurde innerhalb des Programmes `RecordLinkage<1>`, sowie der assoziierten Klasse `RecordLinkageInput<2>` umgesetzt.

Als Input dienten diesem System jeweils zwei Datensätze, die bereits standardisierte, einwegverschlüsselte Kontrollnummern von identifizierenden Daten (IDAT) beinhalteten. Das Format dieser Daten musste dem Rückgabeformat des Programmes `GenerateControlNumbers<6>` entsprechen, das zugrunde liegende Personendaten gemäß Regelvorgaben aus UNICON [55] (siehe Kapitel 1.3.2) erst standardisiert und dann mithilfe der Hash-Funktion SHA-2 (256-Bit) [62] einwegverschlüsselt.

*Tabelle 6* beschreibt die in dieser Arbeit genutzten identifizierenden Basisdaten wie auch die hierauf basierenden standardisierten, einwegverschlüsselten Kontrollnummern so wie sie von der Klasse `GenerateControlNumbers<6>` erzeugt werden.

*Tabelle 6: In dieser Arbeit zur Gewichtsberechnung genutzte IDAT.*

<table>
<thead>
<tr>
<th>IDAT</th>
<th>Segmentierung in Kontrollnummern.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nachname</td>
<td>NACHNAME1, NACHNAME2, NACHNAME3</td>
</tr>
<tr>
<td>Vorname</td>
<td>VORNAME1, VORNAME2, VORNAME3,</td>
</tr>
<tr>
<td>Geburtsdatum</td>
<td>GEBURTSTAG, GEBURTSMONAT, GEBURTSJAHR</td>
</tr>
<tr>
<td>PLZ</td>
<td>PLZ</td>
</tr>
<tr>
<td>Wohnort</td>
<td>ORT</td>
</tr>
<tr>
<td>Geschlecht</td>
<td>GESCHLECHT</td>
</tr>
<tr>
<td>Personen-Identifikationsnummer</td>
<td>PID</td>
</tr>
</tbody>
</table>

 Während des Standardisierungsschrittes wurden zudem eine Reihe von Kontrollnummern, die ausschließlich als Blocking-Variablen dienten, erzeugt. Hierbei handelte es sich um den phonetischen Nachnamen, den phonetischen Vornamen sowie das Geburtsdatum (siehe *Tabelle 7*).

*Tabelle 7: Blocking-Variablen inklusive der IDAT, aus der die BV generiert wurden.*

<table>
<thead>
<tr>
<th>IDAT</th>
<th>Blocking-Variablen</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nachname</td>
<td>PHO_NACHNAME</td>
</tr>
<tr>
<td>Vorname</td>
<td>PHO_VORNAME</td>
</tr>
<tr>
<td>Geburtsdatum</td>
<td>GEBURTSDATUM</td>
</tr>
</tbody>
</table>
Material und Methoden


**Abbildung 9: Schematischer Ablauf des für diese Arbeit verwendeten Record-Linkage-Systems.**

2.1.3. **Beschreibung der verwendeten klinischen Daten**

Für diese Arbeit wurde ein realer Datensatz, bestehend aus Personen identifizierenden Daten zu 46.629 Patienten des Klinikums Großhadern (http://www.klinikum.uni-muenchen.de) verwendet. Die Patientendaten wurden dabei im Vorfeld anhand des Programmes *GenerateControlNumber* standardisiert und einwegverschlüsselt. Es handelte sich dabei um eine zufällige Stichprobe aus einer Gesamtmenge von insgesamt 466.286 Patienten, die in den Jahren 2008-2012 im Klinikum zur Behandlung registriert wurden (dieser Datensatz enthielt keine Daten von Patienten, deren Aufnahme storniert wurde). Der zur Verfügung gestellte
Datensatz entsprach somit einem Anteil von ca. 10% der Patienten, die während des genannten Zeitraumes tatsächlich behandelt wurden.

Durch die Größe des Datensatzes sollte eine relativ bevölkerungsnahe und realistische Verteilung von Attributen wie beispielsweise Vornamen oder Nachnamen in der Region zu erwarten sein. Dadurch, dass die meisten Patienten spekulativ aus dem Großraum München und Umgebung stammen sollten, war zu erwarten, dass der Datensatz im Gegensatz zu komplett künstlichen Datensätzen zudem interessante Verwandtschaftsbeziehungen wie etwa das Vorkommen von Zwillingen enthielt, die in der Regel hohe Anforderungen an ein Record-Linkage stellen.

2.2. Überwachte Klassifizierung – angestrebtes Vorgehen

Im Zuge dieser Arbeit galt es unter anderem, ein überwachttes Klassifizierungssystem zu entwickeln und mit unüberwachten Klassifikationstechniken abzugleichen. Dieses überwachte System sollte dabei, angepasst an die Originaldaten, Trainingssets konstruieren auf denen ein optimaler Trainingsset-spezifischer Klassifikator ermittelbar wäre welcher schließlich als Klassifikator auf den Originaldaten verwendet werden könnte. Die genaue Konstruktion der Trainingssets in Bezug auf die einzelnen Konstruktionsparameter wie beispielsweise die Größe der Teilsets sollte innerhalb dieser Arbeit ermittelt, und auf beste Performanz (Abgleichsgüte) hin optimiert werden (siehe Kapitel 2.5 bzw. 2.6). Der generelle Ablauf der angestrebten überwachten Klassifizierungsmethodik konnte aber bereits spezifiziert werden und unterteilte sich in folgende Schritte (siehe auch Abbildung 10):

5. Die Originaldaten werden per Record-Linkage abgeglichen.
6. Der in (4) berechnete Klassifikator dient als unüberwachter Klassifikator auf den Originaldaten.

![Diagram](image_url)

**Abbildung 10: Konzept zur angestrebten überwachten Klassifizierungsmethodik.**

2.3. Erzeugung von Testsets anhand klinischer Daten

2.3.1. Notwendigkeit der Testset-Erzeugung


Abbildung 11: Darstellung eines für im Kontext des Record-Linkage nutzbaren Testsets.

Man bezeichnet diese Teilmenge auch als Menge der echten Übereinstimmungen (Matches). Im weiteren Verlauf der Arbeit wurden die größeren der beiden Teilsets eines Datensets jeweils als Teilset A und die kleineren der Teilsets als Teilset B bezeichnet. Nach einem Record-Linkage-Durchlauf ließ sich die Güte des Record-Linkage anhand der Diskrepanz der Übereinstimmungen und Nicht-Übereinstimmungen zu den als echt bzw. falsch klassifizierten Links berechnen (siehe Kapitel 1.3.5).

2.3.2. Spezifizierung der Parameter zur Testset-Erzeugung

Material und Methoden

Tabelle 8: Ausprägungsliste der Konstruktionsparameter.

<table>
<thead>
<tr>
<th>Größe</th>
<th>Überlappung</th>
<th>Qualitätsstufe</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>5%</td>
<td>1-10</td>
</tr>
<tr>
<td>1000</td>
<td>25%</td>
<td></td>
</tr>
<tr>
<td>10000</td>
<td>50%</td>
<td></td>
</tr>
<tr>
<td>25000</td>
<td>75%</td>
<td></td>
</tr>
</tbody>
</table>

\[
|Testsets| = \frac{(|Größe| + |Teilsets| - 1)!}{(|Größe| - 1)! |Teilsets|!} \times |Überlappung| \times |Beschaffenheit|
\]

\[
= \frac{(4 + 2 - 1)!}{(4 - 1)! 2!} \times 4 \times 10 = \frac{5!}{3! 2!} \times 4 \times 10 = 10 \times 4 \times 10 = 400
\]


Für den weiteren Verlauf der Arbeit war es wichtig, die Kenntnis zur genutzten Parametrisierung der Testdaten zu dokumentieren. Dies geschah direkt über den Dateinamen (siehe Abbildung 13).

![Screenshot des Projektverzeichnisses](image)

Abbildung 12: Ausschnitt aus dem Projektverzeichnis der Programmierumgebung.

Die numerischen Werte standen hierbei stellvertretend für die in Tabelle 9 beschriebenen Ausprägungen.

Tabelle 9: Kodierung der Testset-Benennung. (siehe Abbildung 13)

<table>
<thead>
<tr>
<th>Größenkombination</th>
<th>Überlappung</th>
<th>Beschaffenheit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vermerk im Dateiname</td>
<td>Wert</td>
<td>Vermerk im Dateiname</td>
</tr>
<tr>
<td>0</td>
<td>[100:100]</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>[100:1000]</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>[100:10000]</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>[100:20000]</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>[1000:1000]</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>[1000:10000]</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>[1000:20000]</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td>[10000:10000]</td>
<td>7</td>
</tr>
<tr>
<td>8</td>
<td>[10000:20000]</td>
<td>8</td>
</tr>
<tr>
<td>9</td>
<td>[20000:20000]</td>
<td>9</td>
</tr>
</tbody>
</table>

2.3.3. Konkrete Implementierung der Testset-Erzeugung

Erzeugung von Teilset A

Material und Methoden

Abbildung 14: Erzeugung individueller Testsets basierend auf unterschiedlicher Parametrisierung.

**Erzeugung von Teilset B**

**Auffüllen des Teilsets abzüglich des Überlappungsbereiches**

Die Erstellung der kleineren Teilsets, die weiterführend jeweils als Teilset B bezeichnet wurden, stellte sich als etwas komplexer dar. Das jeweilige Teilset B wurde gemäß des gegebenen Größenparameters aus demselben Topf an noch verbleibenden Klinikumsdaten, aus dem bereits Teilset A erstellt wurde, mit Patienten aufgefüllt. Zu beachten war allerdings, dass der Anteil der Überlappung in diesem Teilset B zu diesem Zeitpunkt noch nicht belegt wurde (siehe Abbildung 14b).

**Erstellen des Überlappungsbereiches**

Zu dem noch nicht befüllten Überlappungsbereich wurden nun Patienten (ohne Duplikate) aus dem Teilset A in das Teilset B kopiert. Der Überlappungsbereich enthielt somit die Patienten, die sowohl in Teilset A als auch in Teilset B auftraten und die über die gleich bleibende PID in beiden Datensätzen erkennbar waren (siehe Abbildung 14c).
Ohne weitere Bearbeitung wäre dieser Überlappungsbereich nun durch ein Record-Linkage problemlos zu identifizieren gewesen, da es sich um direkte Kopien, also 100%ige Übereinstimmungen in den Attribute zwischen den Patienten der beiden Teilsets handelte. Die Testsets dienten jedoch dem Zweck, realistische Szenarien so gut wie möglich zu simulieren. Aus diesem Grund wurden die Attribute der Patienten im Überlappungsbereich gemäß dem Beschaffenheitsparameter des jeweiligen Testsets verunreinigt bzw. mit Fehlern versehen.

**Einfügen von Fehlern in Kontrollnummern der Patienten innerhalb des Überlappungsbereiches**

Während dieses Schrittes wurden Fehler entsprechend der durch die einzelnen Beschaffenheitsstufen (1 bis 10) definierten Fehlerhäufigkeiten in die Kontrollnummern der Patienten im Überlappungsbereich übertragen. Die verwendeten Fehlerhäufigkeiten leiteten sich hierbei aus zwei Berichten ab, zum einen aus einem Bericht aus dem Krebsregister NRW [73], zum anderen zu generell empfohlenen Schätzwerten der m-Werte während eines Record-Linkage (also den invertierten Fehlerhäufigkeiten) in Krebsregistern [39]. Anhand der beiden Referenzen wurden hierbei die Beschaffenheitsstufen 1 bzw. 2 erstellt, die eine gute Datenqualität, so wie sie in gepflegten Registern vorkommen sollte, darstellen sollten. Die Differenz in den attributabhängigen Fehlerwahrscheinlichkeiten zwischen Beschaffenheitsstufe 1 und Beschaffenheitsstufe 2 wurde verwendet, um die Fehlerhäufigkeiten in den restlichen Beschaffenheitsstufen (3-10) zu ermitteln. Die Beschaffenheitsstufe 10 stellte somit Testsets mit der niedrigsten Datenqualität dar. Die genauen Fehlerhäufigkeiten, abhängig von der Beschaffenheitsstufe, werden in Tabelle 10 bzw. Abbildung 15 wiedergegeben.

_Tabelle 10: Fehlerhäufigkeiten abhängig von Qualitätsstufe und Attributsgruppe_

<table>
<thead>
<tr>
<th>Attributsgruppe</th>
<th>Konkrete Attribute</th>
<th>Fehlerquote nach Beschaffenheit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Start</td>
</tr>
<tr>
<td>Namensattribute</td>
<td>NACHNAME1, NACHNAME2, NACHNAME3, VORNAME1, VORNAME2, VORNAME3</td>
<td>0,025</td>
</tr>
<tr>
<td>Datumsangaben</td>
<td>GEBURTSTAG, GEBURTSJahr</td>
<td>0,01</td>
</tr>
<tr>
<td>Adressangaben</td>
<td>PLZ, ORT</td>
<td>0,05</td>
</tr>
<tr>
<td>Geschlecht</td>
<td>GESCHLECHT</td>
<td>0,001*</td>
</tr>
</tbody>
</table>
Der Startwert gibt die initialen Fehlerhäufigkeiten in den einzelnen Attributsgruppen bei einer Beschaffenheitsstufe 1 wieder. Für jede Beschaffenheitsstufe erhöhte sich die Fehlerhäufigkeit um einen attributsspezifischen Faktor, der, wie erwähnt, der Differenz aus Q1 und Q2 entsprach.

Abbildung 15: Mögliche Fehlerhäufigkeiten in Testsets abhängig von Qualitätssstufe und Attributsgruppe.

Grundsätzlich gibt es verschiedene Vorkommen von Fehlern, die in den verschiedenen Attributsgruppen verschieden häufig vorkommen. Diese wurden nach eigenem Ermessen wie folgend spezifiziert:

- **Deformationsfehler**: Fehler, die eine Ausprägung in eine nicht valide Ausprägung umwandeln.
- **Transformationsfehler**: Fehler, die eine Ausprägung in eine andere valide Ausprägung umwandeln.
- **Fehlender Wert**: Die Entität besitzt für dieses Element keine Ausprägung.


Bei Fehlern im Feld Geschlecht, bzw. in Datumsangaben handelt es sich meist um Transformationsfehler. Dies heißt, eine Attributsausprägung wird in eine tatsächlich vorkommende andere Ausprägung umgewandelt. Auch im Namen und den Adressfeldern dürfte die Mehrzahl der Fehler auf Transformationsfehler zurückzuführen sein. Als Beispiel sei


Zu den Häufigkeiten der vorkommenden Fehler in medizinischen Daten konnten keine Angaben gefunden werden. Die Fehlerhäufigkeiten wurden aus diesem Grund heuristisch, also basierend auf eigenen Erfahrungen, geschätzt (siehe Tabelle 11 sowie Abbildung 16).

Tabelle 11: Häufigkeit von Fehlerarten in Abhängigkeit der gegebenen Attributsgruppe.

<table>
<thead>
<tr>
<th>Attributsgruppe</th>
<th>Transformation</th>
<th>Deformation</th>
<th>Fehlender Wert</th>
</tr>
</thead>
<tbody>
<tr>
<td>Namensattribute</td>
<td>70%</td>
<td>20%</td>
<td>10%</td>
</tr>
<tr>
<td>Datumsangaben</td>
<td>80%</td>
<td>0%</td>
<td>20%</td>
</tr>
<tr>
<td>Adressangaben</td>
<td>40%</td>
<td>30%</td>
<td>30%</td>
</tr>
<tr>
<td>Geschlecht</td>
<td>70%</td>
<td>5%</td>
<td>25%</td>
</tr>
</tbody>
</table>

Entsprechend der gegebenen Häufigkeiten wurden nun Fehler in die Kontrollnummern der Patienten innerhalb des Überlappungsbereiches des kleineren Teilsets eingefügt. (siehe Abbildung 14d). Hierbei wurde für jede Attributsausprägung ein zufälliger Fließkommawert zwischen 0 und 100 generiert und mit den gegebenen Fehlerraten abgeglichen. Lag der Wert unter dem gegebenen Schwellwert wurde ein Fehler nach nachfolgendem Schema erzeugt.
Material und Methoden

Abbildung 16: Häufigkeit der Fehlerart in Abhängigkeit der gegebenen Attributsgruppe.

Bei Deformationsfehlern wurde ein zufälliges Symbol in den Hash-Werten durch ein nicht im Hexadezimalcode vorkommendes Zeichen ersetzt. Hierdurch entstanden neue deformierte Werte, die in dieser Form außer bei Auftreten des exakt selben Fehlers bisher nicht in der Wertemenge enthalten waren.

Bei Auftreten von Transformationsfehlern wurde die alte Ausprägung durch eine neue aus der Gesamtwertemenge der Klinikumsdaten stammende Ausprägung ersetzt.

Bei fehlenden Werten wurde der alte Hash-Wert durch einen leeren String ersetzt.

Übertragung der Matches in das Teilset B

Der mit Fehlern versehene Überlappendungsbereich konnte nun an die bereits bestehende Liste an Einträgen in Teilset B angehängt werden (siehe Abbildung 14 e).

2.3.4. Auswertung der Testsets

In den nachfolgenden Analysen (siehe Kapitel 2.6) galt es unter anderem, die Güte verschiedener binärer Klassifikatoren auf den 400 gegebenen Testsets zu prüfen. Hierbei war nicht nur der Vergleich der Klassifikatoren untereinander interessant, sondern auch die Information, wie nahe sich diese Klassifikatoren mit ihrer Vorhersage qualitativ an die auf dem jeweiligen Testset bestmögliche Güte annähern konnten. Es galt also, initial zu jedem Testset
die bestmögliche Güte zu bestimmen. Definiert wurde diese in dieser Arbeit als der vom
ejeweiligen Testset abhängige maximale F-Measure-Wert, der durch eine binäre Klassifikation
auf dem ausgewählten Testset erzielt werden kann. Die nachfolgenden Unterkapitel erläutern,
wie bei der Bestimmung der testsetspezifischen, maximalen F-Measure-Werte vorgegangen
wurde.

**Record-Linkage auf den Testsets**

Zu jedem der 400 Testsets wurde mithilfe des unter Kapitel 2.1.2 beschriebenen Systems ein
Record-Linkage durchgeführt. Somit wurden 400 testsetabhängige Gewichtsdateien erzeugt,
auf denen weiterführend der jeweils bestmögliche F-Measure-Wert berechnet werden konnte.
Die Automatisierung des Record-Linkage auf den 400 gegebenen Testsets wurde mithilfe des
Programmes CreateTestSetsWeights<8> realisiert.

**Bestimmung des optimalen F-Measure-Wertes**

Zu den testsetspezifischen Gewichtsdateien wurde der jeweils höchstmögliche F-Measure-
Wert berechnet. Der Algorithmus hierzu war trivial. Zu einem Schrankenwert, der die
Gewichtsdatei in echte und falsche Links unterteilte, ließen sich jeweils anhand der bekannten
ID Übereinstimmungen zwischen Teilset A und Teilset B die TP, FP sowie FN berechnen. Aus
diesen Bemessungen ließ sich zum gegebenen Schrankenwert jeweils der F-Measure-Wert
berechnen. Angefangen beim niedrigsten in der jeweiligen Gewichtsdatei anfangenden
Gewichtswert wurde diese Schranke inkrementell um einen Wert von jeweils 0,1 in Richtung
höherer Gewichte verschoben. An jeder Position erfolgte eine Berechnung des F-Measure-
Werts. Der Maximalwert wurde gespeichert und in eine Datei geschrieben.

Das Inkrement von 0,1 hätte grundsätzlich auch kleiner gewählt werden können, um eine noch
genauere Messung zu gewährleisten, resultierte aber in einer dem Faktor entsprechenden
linearen Laufzeit-Erhöhung der Prozedur. Für diese Arbeit erschien eine Approximation auf
eine Nachkommastelle jedoch ausreichend. Somit muss dem Leser an dieser Stelle klar sein,
dass es theoretisch auch höhere Maximalwerte für den F-Measure-Wert gäbe, was jedoch nur
dann der Fall wäre, wenn mehrere Links ein unterschiedliches Gewicht innerhalb eines
Gewichtsintervales von 0,1 besäßen.

Die konkrete Implementierung hierzu fand sich im Programm FMeasure<9>, mittels dessen
die automatisierte Berechnung des F-Measure-Wertes auf allen 400 gegebenen
Gewichtsdateien durchgeführt wurde. Die Ergebnisse wurden dabei gesammelt in eine Datei
übertragen. Weitere Optimierungsverfahren hierzu wären denkbar.
**Bestimmung der optimalen Schranke**

In der Praxis findet sich oft ein optimaler F-Measure-Wert, der sich nicht nur auf eine Gewichtsposition beschränkt, sondern ein größeres Gewichtsintervall abdecken kann. *Abbildung 3a* verdeutlicht diesen Fakt. Der optimale F-Measure-Wert ist hierbei zwischen den beiden Erhebungen zu erwarten, eine Klassifikationsschranke würde also unabhängig von der Position innerhalb des Intervalls zwischen den beiden Erhebungen im selben F-Measure resultieren. Für die überwachte Klassifizierung, die in den nachfolgenden Kapiteln näher vorgestellt wird, musste jedoch auf Trainingsdaten ein exakter Schrankenwert zum gegebenen maximalen F-Measure-Wert bestimmt werden, der später auf den Testdaten als Klassifikator verwendet werden konnte. Die Festlegung dieses Wertes wurde wie nachfolgend gehandhabt:

1. Gibt es ein Gewichtsintervall, über das sich der maximale F-Measure-Wert streckt, so wird als optimaler Schrankenwert der Mittelwert dieses Intervalls spezifiziert.
2. Gibt es mehrere Intervalle dieser Art, so wird das breiteste Intervall zur Ermittlung der Schranke, gewählt und Regel 1 wird auf dieses Intervall angewandt.

**Graphische Auswertung in Bezug auf die Parametrisierung**

Die individuellen Testsets wurden anhand der Kombination verschiedener Konstruktionsparameter erzeugt. Interessant war es hierbei, ob und inwiefern die verschiedenen Konstruktionsparameter einen Einfluss auf die bestmögliche Klassifikationsqualität besaßen.


**2.4. Identifikation von potentiell einflussreichen Parametern auf die Erzeugung von Trainingssets**

Um die Parametrisierung der Trainingsset in Bezug auf überwachte Klassifizierung zu normieren, und um hierbei ein mögliches Optimum zu ermitteln, wurde zu dieser Arbeit folgende Hypothese aufgestellt:

*Je ähnlicher ein Trainingsset dem zu prüfenden Testset ist, umso ähnlicher sind auch deren optimale Klassifikatoren.*

Die Interpretation hierzu lautete: Konstruktionsparameter, wie beispielsweise die Größe der Teilsets, die zur Konstruktion von Trainingssets verwendet wurden, sollten denen der Ausgangsdaten möglichst entsprechen.


Material und Methoden

Trainingsdaten auf die Größe des Überlappungsbereiches in den Testdaten ebenfalls eine positive Auswirkung? War es notwendig, die Fehlerraten im Überlappungsbereich der Trainingsdaten möglichst an die der Testdaten anzupassen? War es überhaupt sinnvoll, sich direkt an den Originaldaten zu bedienen, also die Werteverteilung der Trainingsdaten an denen der Testdaten möglichst zu orientieren?

Eine Überprüfung, ob die genannte Hypothese korrekt war und wie sie methodisch interpretiert werden konnte, war Teilauflage dieser Arbeit.

Zu den genannten Parametern, Größe der Teilsets, Größe des Überlappungsbereiches, Fehlerraten im Überlappungsbereich, sowie die Werteverteilung sollten deswegen nachfolgend Untersuchungen vorgenommen werden, um zu prüfen, ob sich eine Anpassung dieser Werte an die Originaldaten positiv auf die Klassifikation eines probabilistischen Record-Linkage-Systems auswirkten oder nicht. Sollte dies für alle der genannten Parameter der Fall sein, wäre die zuvor aufgestellte Hypothese bestätigt.

Nicht geprüft wurden die Anpassung der Domäne bzw. der Datenstruktur an die Trainingsdaten. Es erschien offensichtlich, dass beispielsweise eine Erhöhung der Attributsanzahl in den Trainingsdaten zu einer durchschnittlich höheren Gewichtung von Datenvergleichen führen würde, was in Bezug auf eine möglichst übereinstimmende Klassifikation zwischen Trainings- und Testdaten kontraproduktiv gewesen wäre. Aus diesem Grund wurden in den folgenden Analysen stets Trainingssets mit übereinstimmender Datenstruktur aus derselben Domäne (Patientendaten) verwendet.

2.5. Überprüfung des Einflusses von Konstruktionsparametern auf die Qualität der Klassifikation

2.5.1. Zielsetzung der Parameterprüfung

In den nachfolgenden Kapiteln sollte geprüft werden, ob eine Anpassung der unter Kapitel 2.4 identifizierten, zur Konstruktion der Trainingssets genutzten Parameter an die Ausgangsdaten tatsächlich zu einer verbesserten überwachten Klassifizierung führte. Sollte sich zeigen, dass die Anpassung aller identifizierten Parameter einen positiven Einfluss auf die Klassifizierung ausübte, wäre dies ein Indiz für die Hypothese aus Kapitel 2.4. Unabhängig davon sollte aber versucht werden, die Klassifikationsqualität durch eine Bestimmung passender
Parameterwerte zu maximieren und eine hierauf basierende Methodik zur überwachten Klassifikation bei probabilistischen Record-Linkage-Systemen zur Verfügung zu stellen.

Hierfür sollten zu jedem Testset als Template-Trainingsset bezeichnete Datensets erstellt werden. Diese sollten entsprechend der Hypothese aus Kapitel 2.4 mit möglichst hoher Ähnlichkeit zu den Orginal-Trainingssets erstellt werden. Bei der Konstruktion sollten also die Größe der Teilsets, die Größe des Überlappungsbereiches, Fehlerraten sowie die Verteilungswerte möglichst zwischen Template-Trainingsset und Testset übereinstimmen. Die genaue Konstruktion wird unter Kapitel 2.5.2 näher erläutert.


2.5.2. Erstellen von Template-Trainingssets


Die automatisierte Erzeugung der 400 auf den Testsets beruhenden Template-Trainingssets wurde mithilfe des Programmes CreateTemplateTrainingsset<10> realisiert. Nachfolgend wurden die jeweils einzelnen Teilsets der 400 Template-Trainingssets per Record-Linkage (CreateTrainingSetsWeights<11>) abgeglichen, was in 400 Gewichtsdateien resultierte.

Zu jeder dieser Template-Gewichtsdateien wurde schließlich mit Hilfe der Programme MassFMeasures<12> analog zu Kapitel 2.3, erst ein maximaler F-Measure-Wert und anschließend jeweils ein hierauf basierender optimaler Template-Schrankenwert bestimmt. Dieser vorhergesagte Template-Schrankenwert wurde nun wiederum als Klassifikator, also als Schrankenwert für das jeweilige Testset, wieder verwendet und dessen Qualitätsgüte auf den
Testdaten (F-Measure) dokumentiert. Der Name des hierzu verwendeten Programmes lautet\textit{FitBorderToTestset}13.\textit{<13>}. 

\textit{Abbildung 17: Schematischer Ablauf des Performanzvergleichs zwischen Klassifikatoren eines Template-Trainingssets und einer Trainingsset-Variante.}
Die Template-Trainingssets setzten im Grunde genommen das exakte Wissen über die Parametrisierung (hier über den Dateinamen gegeben) voraus. Im Echteinsetz wären diese Parameter jedoch nicht ohne weiteres exakt ermittelbar. Eigentlich würde es sich um das sogenannte Henne-Ei-Problem handeln [98]. Um den Überlappungsbereich zu bestimmen, bzw. durch einen Klassifikator abzugrenzen, hätte die Größe des Überlappungsbereiches im Vorfeld bekannt sein müssen, was zwar auf Testdaten gegeben war, auf Realdaten jedoch nicht. Als Abhilfe hätte es zu diesem Beispiel theoretische Möglichkeiten gegeben, die Größe des Überlappungsbereiches grob abzuschätzen [71]. Es wären allerdings weitere Untersuchungen über die Qualität dieser Abschätzungen und Auswirkungen auf eine Klassifizierung, die auf Template-Trainingssets beruht, erforderlich gewesen.

2.5.3. **Variation der Größe**

Zu jedem Testset wurden zusätzlich jeweils zwei Trainingsset-Varianten erstellt, die in der Größe von der Template-Parametrisierung abwichen. Hierbei galt es zu prüfen, ob die Klassifikationsqualität abwich, wenn nicht die exakten Größen der Testdaten zur Konstruktion der Trainingsdaten verwendet wurden.


Bei der zweiten Größenvariante wurde die Größe der Teilsets wie bei der ersten Variante nicht auf einen konstanten Wert festgelegt, sondern die Variante orientierte sich an den ursprünglichen Größenverhältnissen im Testset. Die Größe wurde hierbei jeweils halbiert, das Größenverhältnis blieb also erhalten. Trainingssets dieser Variante wurden mithilfe des Programmes CreateSizeVariant2Trainingsset<15> erzeugt.

2.5.4. **Variation der Fehlerrate**

Zur Prüfung, ob die Erhaltung der exakten Fehlerrate bei der Konstruktion der Trainingssets eine Rolle spielte, wurde eine Trainingsset-Variante konstruiert, bei der komplett auf Fehler im Überlappungsbereich verzichtet wurde. Trainingssets dieser Variante wurden mithilfe des Programmes CreateErrorVariantTrainingsset<16> erzeugt.

2.5.5. **Variation der Überlappung**

Um zu prüfen, inwiefern die Klassifikationsqualität bei Variation der Größe des Überlappungsbereiches von der Klassifikation bei Verwendung des Template-Trainingssets abwich, wurde in dieser Variante für die Größe des Überlappungsbereiches nicht der
Originalwert der Testdaten sondern ein fester Standardwert verwendet. Hierbei wurden zu allen der 400 Testdatensätze jeweils drei Varianten entworfen mit festen Standardwerten von jeweils 3%, 30% sowie 90% in Bezug auf die Anzahl von Patientendaten innerhalb des Überlappungsbereiches. Die Prozentzahlen bezogen sich, wie bereits unter Kapitel 2.3.2 beschrieben, auf das jeweils kleinere Teilset. Trainingssets dieser Varianten wurden mithilfe der Programme CreateOverlapVariant1Trainingsset<17>, CreateOverlapVariant2Trainingsset<18> sowie CreateOverlapVariant1Trainingsset<19> erzeugt.

2.5.6. Variation der Verteilung

Letztdlich wurde geprüft, ob es Sinn macht, die Verteilung der Werte in Testsets bestmöglich zu erhalten, oder ob die Werte verteilt eine eher vernachlässigbare Rolle bei der Klassifizierung spielte. Rekapitulierend: Bei den Template-Trainingssets war das Trainingsset A jeweils die direkte Kopie des zugrunde liegenden Testsets A. Die Verteilung der Werte stimmte hier also exakt überein. Trainingsset B orientierte sich ebenfalls an den Testdaten, variierte aber im Überlappungsbereich, bei dem es sich um direkte Kopien aus Trainingsset A handelte. Es sollte sich also auch in Trainingsset B um eine zumindest ähnliche Verteilung wie in Teilset B handeln.

Bei der im Folgenden erläuterten, neuen Verteilungsvariante jedoch wurden die Trainingssets nicht wie bisher üblich mit den direkten Kopien aus den zugrunde liegenden Testsets befüllt. Anstelle der Template-Prozedur wurden die Trainingssets dieser Variante mit künstlich assemblierten Patienten belegt.

2.5.7. **Performancevergleich der Klassifikatoren der Trainingsset-Varianten**


\[ |\text{Trainingssets}| = 400 \times 8 \times 3 = 9600. \]  

(16)

Die Ergebnisse hierzu werden unter Kapitel 3.2 näher beschrieben.

2.6. **Vergleich von unüberwachter Klassifizierung mit anderen Klassifikationstechniken**

2.6.1. **Zielsetzung des Klassifikatorenabgleichs**

Letztere Methode wurde bereits mit anderen unüberwachten Klassifikationsmethoden verglichen und konnte hierbei Verbesserungen bei der Zuordnungsqualität im Bereich des Record-Linkage erzielen. Beispielsweise übertrifft die genannte Methode den Hybrid-TAILOR Ansatz, von dem wiederum gezeigt wurde, dass dieser andere aus dem maschinellen Lernen bekannte Klassifikationsmethoden, was die finale Abgleichsqualität angeht, übertrifft [93].


2.6.2. Überwachte Klassifizierung der Testdaten

Zu jedem der 400 Testsets wurde entsprechend den Erkenntnissen aus Kapitel 3.2 jeweils ein parameter-optimiertes Trainingsset erzeugt. Dieser Vorgang wurde dreimal wiederholt. Der Grund hierfür war, dass somit zu jedem Testset mehrere auf überwachter Klassifizierung basierende Klassifikatoren zur Verfügung standen. Bei der Wahl eines Mittelwertes dieser Klassifikatoren kann also der maximal mögliche Fehler minimiert werden.


2.6.3. Unüberwachte Klassifizierung der Testdaten

Single-Linkage-Clustering

Die Auswahl einer Clustering-Methode sollte zeigen, ob es möglich war, gute Klassifizierungen anhand nicht auf das Record-Linkage speziell angepasster und leicht zu implementierender Klassifizierungsverfahren zu erhalten. Für den Praxisgebrauch wäre dies von Vorteil, da kompliziertere Algorithmen wie beispielsweise SNN-Klassifikation für die meisten Projekte nur

**Seeded-Nearest-Neighbour-Klassifikation**

Material und Methoden

\[ r = \frac{\min(|A|, |B|)}{|W| - \min(|A|, |B|)} \quad (17) \]

\(|W|\) steht hierbei für die Anzahl der Gewichte, \(|A|\) für die Größe des Teilsets A sowie \(|B|\) für die Größe des Teilsets B. Bei der Rückgabe-Variable \(r\) handelt es sich um das Größenverhältnis zwischen der positiven und der negativen Keimmenge. Die negative Keimmenge wurde in dieser Arbeit, vergleichbar zur Veröffentlichung von Peter Christen, auf 5% der Anzahl der Gewichte festgelegt (befüllt mit den niedrigsten 5% der Gewichte).


**Abbildung 19**: Aufteilung der Menge der Links in positive Keimmenge, negative Keimmenge sowie Menge der bisher unklassifizierten Links.

Nach Bestimmung der Keimmengen konnten die enthaltenen Links nun als Trainingsdaten für den eigentlichen Algorithmus genutzt werden. Für diese Arbeit wurde hierzu der K-Nearest-Neighbour-Ansatz implementiert. Der Algorithmus ließ sich wie folgend zusammenfassen. Ein bisher unklassifizierter Link wurde dann zu einer spezifischen Keimmenge hinzugefügt, wenn es sich bei diesem Link um den Link mit der niedrigsten Distanz zu k Links aus der vereinten

\textbf{Abbildung 20: Beispielhafte Illustration des KNN-Algorithmus mit k=3.}
Manuelle Klassifikation durch Auswertung der Testset-Histogramme

Für die manuelle Klassifikation anhand von Histogrammen wurden mithilfe des Programmes `CreateHistogramms<26>` zu jedem Testset Histogramm-Dateien erzeugt. Für die Klassifikation wurde hierbei eine zur DKFS analoge Darstellung gewählt (siehe Abbildung 4). Eine Schranke wurde dabei manuell gesetzt und der Wert in einer Datei hinterlegt. Eine Übersicht der Histogramme in kleinerem geordnetem Format befindet sich in Anhang F. Ergebnisse
3. Ergebnisse

3.1. Testset-Erzeugung


Final wurden die durchschnittlich maximal erreichbaren F-Measure-Werte, abhängig von der zur Konstruktion verwendeten Qualitätsstufe, berechnet (siehe Abbildung 24). Jeder
Datenpunkt bestand hierbei jeweils aus den Ergebnissen von 40 in der Qualitätsstufe übereinstimmenden Testsets.


vermehrt False-Positives. Die Erhöhung beider Werte wirkte sich verringerd auf den F-Measure-Wert aus.


Abbildung 24: Gemittelter, maximaler F-Measure-Wert in Testsets abhängig von der Datenqualität.
3.2. Auf Trainingsset-Varianten basierende Klassifikationsergebnisse

Abbildung 25: Gemittelte Klassifikationsgüte (F-Measure-Wert) von auf verschiedenen Trainingsset-Varianten basierenden Klassifikatoren, gruppiert nach Qualitätsstufe.

Die Grafik ist in der gegebenen Form nur schwer lesbar. Als eindeutiges Ergebnis zeigt sich jedoch schnell und eindeutig, dass die Trainingsset-Variante („Size1“), bei der die Größe der Teilsets auf 100 normiert wurde, nicht zur Klassifikation geeignet war. Die durchschnittlichen F-Measure-Werte lagen hierbei deutlich weit unter den Ergebnissen der anderen Klassifikatoren. Aus der nachfolgenden Grafik (Abbildung 26) wurde die letztgenannte Trainingsset-Variante entfernt und der Fokus richtete sich auf den Bereich der anderen Varianten.
Abbildung 26: Gemittelte Klassifikationsgüte (F-Measure-Wert) von auf verschiedenen Trainingsset-Varianten basierenden Klassifikatoren, gruppiert nach Qualitätsstufe.

In gegebener Grafik zeigten sich nun deutlich die Unterschiede der einzelnen Trainingssetvarianten. Zwar war die Abweichung zwischen Template-Trainingsset und der zweiten Größenvariante („Size2“) nicht so extrem wie bei der ersten Variante, doch auch das Halbieren der Teilsetgrößen resultierte in vergleichsweise verminderten F-Werten. Beide Ergebnisse deuteten daraufhin, dass eine allgemeine Reduktion der Teilsetgrößen zu verminderten F-Werten führte. Dementsprechend sollte für einen optimalen Klassifikator, wie es bereits bei den Template-Trainingssets Usus war, die Teilsetgröße beibehalten werden.


Grundsätzlich überschnitten sich die Konzepte der Größenvariation und der Verteilungsvariation, da sich eine Anpassung der Größe meist direkt auf die Verteilung auswirkte. Dass eine Abweichung der Verteilung beim probabilistischen Record-Linkage direkten Einfluss auf die Klassifikation hatte, war aber grundsätzlich nachvollziehbar. Schließlich basierte beim probabilistischen Record-Linkage die Gewichtsberechnung auf den unter Kapitel 1.3.2 beschriebenen u-Werten, die sich direkt aus der Häufigkeit von Ausprägungswerten ableiteten.


Abbildung 27: Gemittelte Klassifikationsgüte (F-Measure-Wert) von auf verschiedenen Trainingsset-Varianten basierenden Klassifikatoren gruppiert nach Größe des Überlappungsbereiches.

Abbildung 28: Gemittelte Klassifikationsgüte (F-Measure-Wert) von auf verschiedenen Trainingsset-Varianten basierenden Klassifikatoren gruppiert nach Größe der Teilsets.
3.3. CLARA

Basierend auf den vorgehenden Auswertungen war es möglich, die ursprüngliche Interpretation der Hypothese aus Kapitel 2.4 zu widerlegen und es konnten neue, optimierte Empfehlungen zu den Konstruktionsparametern formuliert werden, die die Klassifikationsqualität im Vergleich zur Template-Variante übertrafen. Hierauf basierte das neu entwickelte CLARA-System. CLARA stand hierbei für Classification for Record-Linkage with Artificial Trainingssets. Tabelle 12 beschreibt die optimierte Konstruktionsparametrisierung des CLARA-Systems im Vergleich zur Konstruktion der zuvor beschriebenen Template-Trainingssets.


<table>
<thead>
<tr>
<th>Konstruktionsparameter</th>
<th>Konfiguration (Template)</th>
<th>Konfiguration (CLARA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Größe der Teilsets</td>
<td>Identische Größenverhältnisse der Teilsets zum zugrunde liegenden Testset.</td>
<td>Entsprechend Template-Trainingsset-Konstruktion</td>
</tr>
<tr>
<td>Größe des Überlappungsbereiches</td>
<td>Identisch zur Größe des Überlappungsbereiches des zugrunde liegenden Testsets.</td>
<td>Möglichst minimal, jedoch ausreichend groß um eine Klassifikation grundsätzlich zu erlauben. Für diese Arbeit und generell als Richtwert werden 3% der Größe des jeweils kleineren Teilsets vorgeschlagen.</td>
</tr>
<tr>
<td>Verteilung</td>
<td>Trainingsset A identisch zu Testset A. Trainingsset B bis auf Überlappungsbereich identisch zu Testset B.</td>
<td>Entsprechend Template-Trainingsset-Konstruktion</td>
</tr>
<tr>
<td>Fehlervorkommen</td>
<td>Häufigkeitswerte zu Fehlervorkommen stimmen mit denen des Testsets überein.</td>
<td>Es werden keine Fehler in den Überlappungsbereich eingebracht.</td>
</tr>
</tbody>
</table>

Ergebnisse

nicht mehr notwendig, da ein konstanter Wert (3%) verwendet wurde. Ebenso waren Schätzungen zu den Fehlerraten unnötig, da diese nach den Ergebnissen aus Kapitel 3.2 nicht mehr benötigt wurden, bzw. der Klassifikation nicht zugute kamen. Man versuchte die Verteilung, wie gehabt, möglichst unverändert zu belassen, was ohne Vorkenntnisse, wie bereits beschrieben, durch einfaches Kopieren aus den Originaldaten möglich war.

Abbildung 29: Schematischer Ablauf der ganzheitlichen CLARA-Methodik.
Durch Erzeugung und Schrankenberechnung mehrerer Trainingssets mit Variation im Überlappungsbereich konnten mehrere CLARA-Klassifikationen zu jeweils einem Testset hervorgesagt werden. Ein Mittelwert dieser multiplen Klassifikatoren würde also den maximal möglichen Fehler, also die Abweichung des Klassifikators vom eigentlichen optimalen Schrankenwert, minimieren, was beim konkreten Einsatz beachtet werden sollte. *Abbildung 29* beschreibt zusammenfassend den schematischen Ablauf des finalen CLARA-Verfahrens.

### 3.4. Vergleich verschiedener Klassifikationsmethoden


*Abbildung 30:* Durchschnittlicher F-Measure-Wert verschiedener Klassifikatoren abhängig von der Datenqualitätsstufe.
Hierbei unterschied sich vor allem der SNN in der zweiten Variante von den restlichen Methoden. Auf Testsets mit einer Qualitätsstufe einschließlich dem Wert Q6 erzeugte der Klassifikator noch gute Ergebnisse, erzielte dabei sogar teils bessere Ergebnisse als die manuelle Klassifikation, brach jedoch ab einem Wert von Q7 in Bezug auf die Abgleichsqualität stark ein.

Abbildung 31: Durchschnittlicher F-Measure-Wert verschiedener Klassifikatoren abhängig von der Datenqualitätsstufe.

Abbildung 32 Durchschnittlicher F-Measure-Wert verschiedener Klassifikatoren abhängig von der Datenqualitätsstufe.
Im direkten Vergleich übertraf die manuelle Klassifikation den SNN. Im Vergleich zum CLARA-System zeigte sich vor allem, dass der SNN nicht nur bei schlechter Datenqualität schlechter als das CLARA-System abschnitt, sondern auch bei guter Datenqualität.

4. Diskussion

4.1. Begründung der Konzeption eines überwachten Klassifikationssystems

Im Rahmen der DKFS wurden wissenschaftlich offene Fragestellungen und Probleme in Bezug auf die Klassifikation im Bereich des Privacy-Preserving-Record-Linkage identifiziert. Aufgrund schwieriger Datenverhältnisse, wie sie sich gerade im Fall der Daten von Angehörigen präsentierten, kann es Probleme bereiten, eine passende Klassengrenze bzw. einen binären Klassifikator zu bestimmen [67].

Zu Problemen dieser Art gibt es nur wenig Literatur, da zum einen wohl die Datengrundlage in vielen Projekten eine einfachere Klassifikation erlaubt. Zum anderen scheint es, als würde die Relevanz der Klassifikation oft im Schatten der Gewichtsberechnung stehen, die in wissenschaftlicher Literatur die meiste Aufmerksamkeit genießt.

Bei automatisierten Klassifikationsmethoden, die während eines anonymen Record-Linkage alternativ zum manuellen Vorgehen anwendbar wären [71], handelt es sich primär um regelbasierte, überwachte sowie unüberwachte Klassifikationssysteme. Während regelbasierte Klassifikationsmethoden meist sehr projektspezifisch aufgesetzt werden, konzentrierten sich die Untersuchungen der Klassifikationsmethoden in dieser Arbeit dagegen vorrangig auf den Vergleich zwischen unüberwachter sowie überwachter Klassifizierung [38,41,71,80].

Insbesondere wurde dabei eine eigens entwickelte, schon früh entworfene Idee zur überwachten Klassifizierung ausgearbeitet, die später mit anderen Klassifikationsmethoden verglichen wurde. Die Fokussierung auf die überwachte Klassifizierung rührte aus der Annahme, dass schlechte Datenqualität eine überwachte Klassifikation weniger negativ beeinflussen sollte als eine unüberwachte Klassifikation, die bei Artefakten in der Gewichtsmenge, wie etwa unerwartete, zufällig auftretende Gewichtssprünge, immer die Gefahr einer kompletten Fehlklassifikation birgt. Aufgrund der Tatsache, dass die manuelle Klassifikation auf Histogramm-Daten ebenfalls dieselben Probleme aufweist – also Anfälligkeit gegenüber Datenartefakten – stellte sich die überwachte Klassifikation als unabhängige Variante hierzu dar [42].

Zwar existieren auch im Bereich des Record-Linkage Ansätze zu überwachter Klassifikation [38,83,84], allerdings fehlen hier eindeutige Anweisungen bzgl. Parametrisierung und Auswahl.
der zugrunde liegenden Trainingssets. Bezüglich des neuen Ansatzes gab es deswegen das Ziel, eine möglichst einfache und eindeutige Anwendung zu erlauben, die im Grunde genommen keine externen Trainingsdaten voraussetzte, sondern die Trainingsdaten direkt aus der zugrunde liegenden Testdatenmenge generierte. Dreh-und Angelpunkt dieser Arbeit war daher, ein derartiges System aufzusetzen und auf verschiedenen Testsets auf die Abgleichsgüte zu prüfen.

4.2. Zugrundeliegende Arbeitsmaterialien


Zu jedem Testdatensatz wurde ein probabilistisches Record-Linkage durchgeführt, wodurch jeweils eine Gewichtsdatei für vergleichende Analysen erzeugt wurde. Das verwendete System entsprach hierbei in Bezug auf die Abgleichsgüte (Sensitivität/Spezifität) anderen aus verschiedener Literatur bekannten Angaben (siehe Tabelle 13 sowie Abbildung 33/Abbildung 34).


<table>
<thead>
<tr>
<th>Quelle</th>
<th>Kurzbeschreibung</th>
<th>Spezifität</th>
<th>Sensitivität</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boonchai et al. [101]</td>
<td>Für eine Prüfung der Qualität eines Record-Verfahrens zwischen zwei künstlichen Datenbanken wurden einwegverschlüsselte Kontrollnummern anhand von Personen-identifizierenden Daten aus verschiedenen Quellen erzeugt und zu Datenbank-Einträgen zusammengefügt.</td>
<td>100%</td>
<td>95%-100%</td>
</tr>
<tr>
<td>Durham et al. [53]</td>
<td>Record-Linkage auf 756.629 künstlichen Patienten-Daten, ausgehend von 100.000 realen Patienten mit einem Überlappungsbereich von 0.01 %.</td>
<td>~100%</td>
<td>~97%</td>
</tr>
<tr>
<td>Quelle</td>
<td>Kurzbeschreibung</td>
<td>Spezifität</td>
<td>Sensitivität</td>
</tr>
<tr>
<td>-----------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------------</td>
<td>--------------</td>
</tr>
<tr>
<td>Contiero et al. [102]</td>
<td>Es wurde ein Abgleich auf einem Teil von Patientendaten des französischen Krebsregisters der Lombardie (20.724 Einträge) mit Daten zu sozialer Sicherheit durchgeführt (1.021.846 Einträge) durchgeführt. Die Ergebnisse wurden über manuelle Kontrolle, also nach Golds-Standard ausgewertet.</td>
<td>98.8%</td>
<td>96.5%</td>
</tr>
<tr>
<td>Fonseca et al. [103]</td>
<td>Die nationale, brasilianische HIV/AIDS Überwachungsdatenbank (559.442 Einträge) wurde gegen eine Menge von 6.444.822 Daten zu registrierten Toden abgeglichen.</td>
<td>99.6%</td>
<td>87.6%</td>
</tr>
<tr>
<td>Migowski et al. [104]</td>
<td>In dieser brasilianischen Studie wurde versucht, die Qualität des Record-Linkage abzuschätzen, indem in einer Datenbank zu verstorbenen Bevölkerung nach am Herzen operierten Patienten gesucht wurde.</td>
<td>100%</td>
<td>90.6%</td>
</tr>
<tr>
<td>Quantin et al. [19]</td>
<td>Abgleich von manueller und automatischer Methodik im Burgundy-Register von Patientendaten mit zum Verdauungssystem assoziierten Krebsarten.</td>
<td>97%</td>
<td>93%</td>
</tr>
<tr>
<td>Fournel et al. [105]</td>
<td>Abgleich des größten französischen Krebsregisters und Todesfällen in Frankreich zwischen 1998–2004.</td>
<td>99.5%</td>
<td>94.8%</td>
</tr>
<tr>
<td>Silveira et al. [75]</td>
<td>Review verschiedener Paper und Studien in Bezug auf Abgleichsqualität von probabilistischem Record-Linkage.</td>
<td>99-100%</td>
<td>74-98%</td>
</tr>
</tbody>
</table>

Wie Abbildung 32 und Abbildung 33 demonstrieren, übertrafen die Werte zu Sensitivität und Spezifität abhängig von der Qualitätsstufe meist sogar die gegebenen Vergleichswerte. Bei Nennung mehrerer Werte in der jeweiligen Arbeit wurde innerhalb der angegebenen Grafiken ein Mittelwert angegeben. Berücksichtigt werden muss hierbei allerdings, dass für das eigene System eine optimale binäre Klassifikation, sowie das Bekanntsein der zugrunde liegenden Häufigkeiten der m-Werte verwendet wurden, was im Realeinsatz nicht der Fall ist und wodurch, mit hoher Wahrscheinlichkeit, eine verbesserte Abgleichsqualität erreicht werden konnte.
4.3. Hypothese als Ausgangspunkt des wissenschaftlichen Vorgehens

Bei der Konzipierung des neuen überwachten Klassifikationsansatzes wurde schließlich initial eine Hypothese aufgestellt, die besagte: Je ähnlicher zwei Datensets sind, umso ähnlicher sind auch ihre Klassifikatoren. In diesem Kontext musste Ähnlichkeit definiert werden und anhand dessen wurde ein Template-Trainingsset konzipiert, das mit dem jeweiligen Testset in Größe der Teilsets, Größe des Überlappungsbereiches, sowie Häufigkeit der Fehlerraten übereinstimmte. Zudem wurde versucht, auch die Werteverteilung möglichst gut zu übernehmen, um die Ähnlichkeit zu maximieren. Es ist nicht auszuschließen, dass es hierbei...
Ansätze gibt, die zu einer noch höheren Ähnlichkeit zwischen Test- und Trainingsdaten führen würden.

Der Hypothese folgend müsste also ein optimaler Klassifikator auf diesem Template-Trainingsset, kalibriert am maximalen F-Measure-Wert, auch auf dem zugrunde liegenden Testdatenset eine Klassifikation mit hoher F-Measure-Bewertung erzeugen. Widersprüchlich wäre es also gewesen, wenn Trainingsdaten, die nicht diesen Ähnlichkeitsanforderungen entsprächen, zu besseren Klassifikationsergebnissen geführt hätten. Um die Annahme also zu prüfen, wurden zu den 400 Testdatensets insgesamt jeweils 7 weitere Trainingsdaten-Varianten aufgesetzt, die sich jeweils in einem Parameter, entweder der Größe der Teilsets, der Größe des Überlappendbereiches, den Fehlerhäufigkeiten, oder der Werteverteilung von den gegebenen Template-Trainingsset unterschieden.


Wie sich zudem zeigte, spielten auch die Häufigkeiten der Fehler in den Überlappendbereichen keine entscheidende Rolle. Diese beeinflussten die Klassifikation weder positiv noch negativ.

Auf die Konstruktion eines optimierten Trainingsdatensets wirkt sich dies natürlich positiv aus, da weder Überlappendbereich, noch Fehler korrekt abgeschätzt werden müssen. Hätte sich herausgestellt, dass diese Parameter denen der Ursprungsdaten entsprechen müssten, wäre die Umsetzung einer Anwendung im Realeinsatz deutlich schwieriger gewesen, da man dann Schätzwerte zu diesen Parametern benötigt hätte. Im Grunde genommen wäre dies das Henne-Ei-Problem, bei dem Werte, die man eigentlich bestimmen will (z.B. die Größe des

4.4. Abgleich und Bewertung verschiedener Klassifikatoren


Wie bereits ausgeführt, konnte die zweistufige Methodik gute Ergebnisse auf Testdaten mit hoher Datenqualität erzielen. Auf Testsets mit mangelnder Datenqualität nahm die Güte der
Klassifikation jedoch rapide ab, da sich mit Abnahme der Datenqualität auch die Häufigkeit von Datenartefakten (unerwartete Abstände, Anhäufungen) erhöht. CLARA übertraf die Klassifikation des genannten Klassifikators auf niedriger Datenqualität bei Weitem, überraschender Weise zeigte sich aber, dass CLARA auch auf Datensätzen mit hoher Datenqualität ähnliche bzw. sogar bessere Ergebnisse als der SNN erzielte.


CLARA übertraf auch die erreichte Klassifikationsgüte der manuellen Schrankenbestimmung anhand von Histogrammen. Dieses Ergebnis würde dafür sprechen, die manuelle Klassifikation komplett durch das CLARA-System zu ersetzen.

Laut Han et Al. gibt es zudem bei überwachten Klassifikationssystemen, wie z.B. CLARA, die Gefahr einer Überanpassung (Overfitting) der Trainingsdaten an die Testdaten, was sich negativ auf die Klassifikationsgüte auswirken könnte [41,80]. Diese Befürchtung war bei der Anwendung von CLARA nicht zu bestätigen. Wie sich anhand der Trainingsset-Varianten zeigte, war das Klassifikationsergebnis immer dann am höchsten, wenn die Verteilung der Ursprungswerte möglichst den Originalwerteverteilungen entsprach. Generell spielt Overfitting für das System keine Rolle da jeder Klassifikator immer für das gegebene Originaltestdaten-Set und nicht für andere Testdatensets einzeln generiert wird. Generell lagen die Klassifikationsergebnisse von CLARA unabhängig von der zugrunde liegenden Datenqualität der Testdatensätze extrem nah am erreichbaren Optimalwert.

4.5. Übertragung der Ergebnisse auf den aktuellen Stand der Wissenschaft

Die Klassifikationsergebnisse von CLARA zeigten auf einer umfangreichen Menge von Testdaten, dass überwachte Klassifikation, repräsentiert durch die CLARA-Technologie, unüberwachter Klassifikation, repräsentiert durch SLC und den SNN, grundsätzlich überlegen war. Eine Auswertung in solch einem Umfang, auf einer Menge von insgesamt 400 individuellen Testdatensätzen, hatte bisher noch nicht stattgefunden [38].


Neben dem Vergleich zwischen unüberwacheter sowie überwachter Klassifikation wäre das Konzept zum CLARA-System an sich als weiterer Beitrag zum Stand der Wissenschaft zu nennen. Das CLARA System baut in dieser Arbeit grundsätzlich auf der Konstruktion von Trainingsdaten, anschließendem Record-Linkage auf diesen Daten, Bestimmung einer Schranke

4.6. Limitierungen der Arbeit


Trotz der auf den Testdaten gegebenen guten Abgleichsgüte gibt es Sonderfälle, mit denen das System nicht gut umgehen kann und die auch hier zu einer starken Fehlklassifikation führen können. Würden etwa per Zufall ausschließlich Links mit einem extrem hohen Abgleichsgewicht (beispielsweise bei doppelten Vornamen) dem Überlappungsbereich

Grundsätzlich handelt es sich bei CLARA außerdem nicht formell um eine überwachte Klassifikation, sondern eher um eine semi-überwachte Klassifikation, da echte Übereinstimmungen, die jedoch nicht bekannt sind, das Ergebnis der vorhergesagten Klassifikatoren eventuell negativ beeinflussen können. Basierend auf den guten Ergebnissen erscheint dieser Einfluss aber nicht mit allzu großen negativen Konsequenzen einherzugehen.

Weitere Einschränkungen wie Laufzeit oder auch benötigter Festplattenspeicher wurden bereits angesprochen, erscheinen jedoch für die meisten Projekte als eher unproblematisch.

5. Zusammenfassung


Bei der neu konzipierten Technik handelte es sich um ein überwachtes Klassifizierungssystem, das die Klassifikatoren anhand von künstlichen Trainingsdaten, die direkt aus den zu vergleichenden Daten generiert wurden, vorhersagte. Entsprechend der Beschreibung wurde das System CLARA benannt (CLAssification for Record-Linkage with Artificial Trainingssets). Die genaue Parametrisierung zur Erzeugung dieser Trainingsdaten wurde über Analysen zu Variationen in den genannten Trainingsdaten optimiert.


Letztendlich lieferten die Analysen Indiz für die Überlegenheit der überwachten Klassifikationssysteme gegenüber den unüberwachten Klassifikationssystemen im Bereich des Record-Linkage. Überwachte Systeme bieten zudem eine von der manuellen Schrankensetzung unabhängige Sichtweise, weswegen diese sehr gut in Kombination verwendet werden könnten.
6. Literaturverzeichnis


37. Palanisamy V, Jeneba M. *Hybrid cryptography by the implementation of RSA and AES. International Journal of Current Research*. April 2011;33(4): 241-244


70. Christen P. A Survey of Indexing Techniques for Scalable Record Linkage and Deduplication. Knowledge and Data Engineerin; 24(9).


82. Christen P. Febrl - a freely available record linkage system with a graphical user interface. HDKM’08, CRPIT vol. 80. 2008.


103. Fonseca M, Coeli C, Lucena F, Veloso V, Carvalho M. *Accuracy of a probabilistic record linkage strategy applied to identify deaths among cases reported to the Brazilian AIDS surveillance database.* 2010.


## 7. Anhang

### A. Abkürzungsverzeichnis

<table>
<thead>
<tr>
<th>Abkürzung</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>AES</td>
<td>Advanced Encryption Standard</td>
</tr>
<tr>
<td>BDSG</td>
<td>Bundesdatenschutzgesetz</td>
</tr>
<tr>
<td>CLARA</td>
<td>CLassification for Record-Linkage with Artificial Trainingssets.</td>
</tr>
<tr>
<td>CLINK</td>
<td>Bezeichnung eines effizienten Complete-Linkage-Clustering Ansatzes</td>
</tr>
<tr>
<td>DKFS</td>
<td>Studie zu familiärem Darmkrebs</td>
</tr>
<tr>
<td>DuDe</td>
<td>The Duplicate Detection Toolkit</td>
</tr>
<tr>
<td>FEBRL</td>
<td>Freely Extensible Biomedical Record Linkage</td>
</tr>
<tr>
<td>FM</td>
<td>F-Measure</td>
</tr>
<tr>
<td>FN</td>
<td>False-Negatives (Falsch Negative)</td>
</tr>
<tr>
<td>FP</td>
<td>False-Positives (Falsch Positive)</td>
</tr>
<tr>
<td>FRIL</td>
<td>Fine-Grained Records Integration and Linkage</td>
</tr>
<tr>
<td>HMAC</td>
<td>Hash-based message authentication code</td>
</tr>
<tr>
<td>IDAT</td>
<td>Identifizierende Daten</td>
</tr>
<tr>
<td>KNN</td>
<td>K-Nearest-Neighbour</td>
</tr>
<tr>
<td>KORA</td>
<td>KOoperative Gesundheitsforschung in der Region Augsburg</td>
</tr>
<tr>
<td>MDAT</td>
<td>Medizinische Daten</td>
</tr>
<tr>
<td>MD5</td>
<td>Message-Digest-Algorithmus (Version 5)</td>
</tr>
<tr>
<td>NSA</td>
<td>National Security Agency</td>
</tr>
<tr>
<td>OYSTER</td>
<td>Open sYSTem Entity Resolution</td>
</tr>
<tr>
<td>PPV</td>
<td>Positive-Predictive-Value (Positiver prädiktiver Wert)</td>
</tr>
<tr>
<td>PRISM</td>
<td>Planning Tool for Resource Integration, Synchronization and Management</td>
</tr>
<tr>
<td>RSA</td>
<td>Rivest, Shamir und Adleman (Initialen der Entwickler)</td>
</tr>
<tr>
<td>SHA</td>
<td>Secure Hash Algorithm</td>
</tr>
<tr>
<td>SLC</td>
<td>Single-Linkage-Clustering</td>
</tr>
<tr>
<td>SLINK</td>
<td>Bezeichnung eines effizienten Single-Linkage-Clustering Ansatzes</td>
</tr>
<tr>
<td>SNN</td>
<td>Seeded-Nearest-Neighbour</td>
</tr>
<tr>
<td>SVM</td>
<td>Support-Vector-Maschine</td>
</tr>
<tr>
<td>TAILOR</td>
<td>RecOrd LInkAge Toolbox (Acronym rückwärts)</td>
</tr>
<tr>
<td>TMF</td>
<td>Technologie- und Methodenplattform für die vernetzte medizinische Forschung</td>
</tr>
<tr>
<td>TN</td>
<td>True-Negatives (Echt Negative)</td>
</tr>
<tr>
<td>TP</td>
<td>True-Positives (Echt Positive)</td>
</tr>
<tr>
<td>TRM</td>
<td>Tumorregister-München</td>
</tr>
<tr>
<td>UNICON</td>
<td>Uniform Control Number Generator</td>
</tr>
<tr>
<td>WHIRL</td>
<td>Word-Based Heterogeneous Information Representation Language</td>
</tr>
</tbody>
</table>
B. Tabellenverzeichnis

Tabelle 1: Unterschiedliche Darstellung einer Entität in zwei verschiedenen Datenbanken. ________ 14
Tabelle 2: Anwendung des SHA-256 auf verschiedene Ausgangswerte. ____________________________ 20
Tabelle 3: Beispielfahste Darstellung des Inhaltes einer Gewichtsdatei. ____________________________ 28
Tabelle 4: Übersicht frei zugänglicher Softwaresysteme im Bereich des Record-Linkage. ___________ 31
Tabelle 5: Wichtigste Hardwarekomponenten des Arbeitssystems. _____________________________ 36
Tabelle 6: In dieser Arbeit zur Gewichtsberechnung genutzte IDAT. ____________________________________________ 37
Tabelle 7: Blocking- Variablen inklusive der IDAT, aus der die BV generiert wurden. _______________ 37
Tabelle 8: Ausprägungsliste der Konstruktionsparameter. __________________________ 44
Tabelle 9: Kodierung der Testset-Benennung. (siehe Abbildung 13) _____________________________ 46
Tabelle 10: Fehlerhäufigkeiten abhängig von Qualitätsstufe und Attributsgruppe ___________ 48
Tabelle 11: Häufigkeit von Fehlerarten in Abhängigkeit der gegebenen Attributsgruppe. _______ 50
Tabelle 12: Beschreibung der Parametrisierung der Konstruktion von Trainingssets des CLARA Systems. 77
Tabelle 13: Angaben zu Spezifität und Sensitivität bzgl. probabilistischem Record-Linkage. _______ 84

C. Formelverzeichnis

Formel 1: Fellegi u. Sunther - Berechnung des Gesamtgewichdes______________________________ 24
Formel 2: Fellegi u. Sunther – Definition: A, B________________________________________________ 24
Formel 3: Fellegi u. Sunther – Definition: M ____________________________________________ 24
Formel 4: Fellegi u. Sunther – Definition: U ____________________________________________ 24
Formel 5: Fellegi u. Sunther – Definition: a, b ____________________________________________ 24
Formel 6: Fellegi u. Sunther – Berechnung des u-Wertes_______________________________ 24
Formel 7: Fellegi u. Sunther – Berechnung des m-Wertes ________________________________________ 25
Formel 8: Fellegi u. Sunther – Gewichtsberechnung bei Übereinstimmung __________________ 25
Formel 10: Dice-Koeffizient ____________________________________________________________ 26
Formel 11: Spezifität ________________________________________________________________ 33
Formel 12: Sensitivität _____________________________________________________________ 33
Formel 13: Positive-Predictive-Measure ________________________________________________ 34
Formel 14: F-Measure-Wert ___________________________________________________________ 34
Formel 15: Berechnung der Anzahl an erstellten Testsets ___________________________________ 44
Formel 16: Berechnung der Anzahl an erstellten Trainingssets ____________________________ 62
Formel 17: Berechnung der Größe der Keimmengen bzgl. Active-Learning Ansatz. __________ 65
D. Abbildungsverzeichnis

Abbildung 1: Datenerfassungsbogen der DKFS. .................................................. 10
Abbildung 2: Vereinfachtes Datenschutz- sowie Datenflussmodell während der DKFS .......... 11
Abbildung 3: Histogramme zur Erläuterung der auftretenden Klassifikationsproblematik. .......... 12
Abbildung 4: Eines der konkreten Histogramme zum Record-Linkage der DKFS am 04.02.2014. ................................................................. 15
Abbildung 5: Pair-Analysis Datei vom Record-Linkage-Durchlauf der DKFS am 19.12.2013. ......... 15
Abbildung 6: Schematischer Ablauf des Privacy-Preserving-Record-Linkage. .......................... 18
Abbildung 7: Einwegverschlüsselung von Werteausprägungen anhand von Bloom-Filtern. ........ 21
Abbildung 8: Kontingenztafel mit dem Urteil der Klassifikation und der tatsächlichen Klasse. ....... 32
Abbildung 9: Schematischer Ablauf des für diese Arbeit verwendeten Record-Linkage-Systems. ...... 39
Abbildung 10: Konzept zur angestrebten überwachten Klassifizierungsmethode. ...................... 41
Abbildung 11: Darstellung eines für im Kontext des Record-Linkage nutzbaren Testsets. ............ 42
Abbildung 12: Ausschnitt aus dem Projektverzeichnis der Programmierungsegebung. ................. 44
Abbildung 13: Automatisierter Ablauf der Testset-Erzeugung. ............................................ 45
Abbildung 14: Erzeugung individueller Testsets basierend auf unterschiedlicher Parametrizierung. .... 47
Abbildung 15: Fehlerhäufigkeiten in Testsets abhängig von Qualitätsstufe und Attributgruppe. ...... 49
Abbildung 16: Häufigkeit der Fehlerart in Abhängigkeit der gegebenen Attributgruppe. ............... 51
Abbildung 17: Performanzvergleich verschiedener Trainingsset-Varianten Klassifikatoren. ......... 58
Abbildung 18: Erzeugung von auf spezifische Testsets angepasste Template-Trainingssets. ......... 59
Abbildung 19: Positive Keimmengen, negative Keimmengen sowie Menge der bisher unklass. Links. . 65
Abbildung 20: Beispielaufgabe für die KNN-Algorithmus. ................................................ 66
Abbildung 21: Ø maximaler F-Measure-Wert in Testsets mit spezifischer Größenkomb. ............... 68
Abbildung 23: Ø maximaler F-Measure-Wert in Testsets bzgl. Überlappung. ............................ 71
Abbildung 24: Ø maximaler F-Measure-Wert in Testsets bzgl. Datenqualität. ............................ 71
Abbildung 25: Ø F-Measure von Trainingsetvarianten-Klassifikatoren gruppiert nach Qualitätsst (1). .... 73
Abbildung 26: Ø F-Measure von Trainingsetvarianten-Klassifikatoren gruppiert nach Qualitätsst (2). .... 74
Abbildung 27: Ø Klassifikationsgüte (F-Measure-Wert) von auf verschiedenen Trainingset-Varianten basierenden Klassifikatoren gruppiert nach Größe des Überlappungsbereiches. ......................... 76
Abbildung 28: Ø Klassifikationsgüte (F-Measure-Wert) von auf verschiedenen Trainingset-Varianten basierenden Klassifikatoren gruppiert nach Größe der Teilsets. ........................................... 76
Abbildung 29: Schematischer Ablauf der ganzheitlichen CLARA-Methodik. ......................... 78
Abbildung 30: Ø F-Measure-Wert verschiedener Klassifikatoren abhängig von der Datenqualitätsstufe. 79
Abbildung 31: Ø F-Measure-Wert verschiedener Klassifikatoren abhängig von der Datenqualitätsstufe. 81
Abbildung 32: Ø F-Measure-Wert verschiedener Klassifikatoren abhängig von der Datenqualitätsstufe. 81
Abbildung 33: Literatur-Vergleich der Spezifität von probabilistischen Record-Linkage-Methoden. .. 86
Abbildung 34: Literatur-Vergleich der Sensitivität von probabilistischen Record-Linkage-Methoden. . 86
E. Programmverzeichnis

<table>
<thead>
<tr>
<th>Index</th>
<th>Programmname</th>
<th>Funktion (Kurzbeschreibung)</th>
<th>Seite</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>RecordLinkage</td>
<td>Hauptklasse zur Durchführung eines Record Linkage auf zwei gegebenen Datensets.</td>
<td>37</td>
</tr>
<tr>
<td>2</td>
<td>RecordLinkageInput</td>
<td>Regelt das Einlesen der Daten für RecordLinkage&lt;1&gt;.</td>
<td>37</td>
</tr>
<tr>
<td>3</td>
<td>Person</td>
<td>Zu RecordLinkageInput&lt;2&gt; assozierte Klasse.</td>
<td>-</td>
</tr>
<tr>
<td>4</td>
<td>ConfigReader</td>
<td>Zu RecordLinkage&lt;1&gt; assozierte Klasse.</td>
<td>-</td>
</tr>
<tr>
<td>5</td>
<td>ListComparator</td>
<td>Zu RecordLinkage&lt;1&gt; assozierte Klasse.</td>
<td>-</td>
</tr>
<tr>
<td>6</td>
<td>GenerateControlNumbers</td>
<td>Klasse zur Standardisierung und Einwegverschlüsselung identifizierender Daten.</td>
<td>37</td>
</tr>
<tr>
<td>7</td>
<td>CreateTestsets</td>
<td>Klasse zur Erzeugung der 400 in dieser Arbeit verwendeten Testdatensätze.</td>
<td>44</td>
</tr>
<tr>
<td>8</td>
<td>CreateTestSetsWeights</td>
<td>Automatisierter Aufruf der Klasse Record Linkage auf den 400 gegebenen Testsets.</td>
<td>52</td>
</tr>
<tr>
<td>9</td>
<td>FMeasure</td>
<td>Berechnung des maximalen FMeasures auf den 400 Gewichtsdateien der Testsets.</td>
<td>52</td>
</tr>
<tr>
<td>10</td>
<td>CreateTemplateTrainingsset</td>
<td>Erzeugung eines Trainingssets unter Verwendung der Konstruktionsparameter eines zugrunde liegenden Testsets.</td>
<td>57</td>
</tr>
<tr>
<td>11</td>
<td>CreateTrainingSetWeights</td>
<td>Erzeugt zu semtlichen Trainingssets die Gewichtsdateien.</td>
<td>57</td>
</tr>
<tr>
<td>12</td>
<td>MassFMeasure</td>
<td>Erzeugt zu den Gewichtsdateien von Trainingssets die FMeasure und Schrankenwerte.</td>
<td>57</td>
</tr>
<tr>
<td>13</td>
<td>FitBorderToTestset</td>
<td>Fügt einen vorhergesagten Klassifikator in ein Testset ein und bemisst den hierdurch erzielten F-Measure-Wert.</td>
<td>58</td>
</tr>
<tr>
<td>14</td>
<td>CreateSizeVariant1Trainingsset</td>
<td>Erzeugung von Trainingssets deren Größe auf 100 festgelegt wurde.</td>
<td>59</td>
</tr>
<tr>
<td>15</td>
<td>CreateSizeVariant1Trainingsset</td>
<td>Erzeugung von Trainingssets deren Größe im Vergleich zu den Testdaten halbiert wurde.</td>
<td>60</td>
</tr>
<tr>
<td>16</td>
<td>CreateErrorVariantTrainingsset</td>
<td>Erzeugung von Trainingssets ohne Fehler im Überlappungsbereich.</td>
<td>60</td>
</tr>
<tr>
<td>17</td>
<td>CreateOverlapVariant1Trainingsset</td>
<td>Erzeugung von Trainingssets deren Überlappungsbereich auf 90% der Größe des kleineren Teilsets festgelegt wurde.</td>
<td>61</td>
</tr>
<tr>
<td>18</td>
<td>CreateOverlapVariant2Trainingsset</td>
<td>Erzeugung von Trainingssets deren Überlappungsbereich auf 30% der Größe des kleineren Teilsets festgelegt wurde.</td>
<td>61</td>
</tr>
<tr>
<td>19</td>
<td>CreateOverlapVariant3Trainingsset</td>
<td>Erzeugung von Trainingssets deren Überlappungsbereich auf 3% der Größe des kleineren Teilsets festgelegt wurde.</td>
<td>61</td>
</tr>
<tr>
<td>20</td>
<td>CreateDistributionVariant1Trainingsset</td>
<td>Erzeugung von Trainingssets in denen die Werteverteilungen der Patienten gleichverteilt wurden.</td>
<td>61</td>
</tr>
<tr>
<td>21</td>
<td>AutomateTrainingsetProduction</td>
<td>Klasse die die Produktion der 9600 Trainingssetvarianten automatisiert.</td>
<td>63</td>
</tr>
<tr>
<td>22</td>
<td>CreateFinalTrainingsset</td>
<td>Trainingsseterzeugung entsprechend dem CLARA Konzept.</td>
<td>63</td>
</tr>
<tr>
<td>23</td>
<td>SingleLinkageNAIV</td>
<td>Vereinfachung des Single Linkage Clusterings. Da es sich bei Gewichtsdateien um eindimensionale Daten handelt ist der Algorithmus trivial und bestimmt die größten Abstände in den Gewichtsdateien als Schrankenwert.</td>
<td>64</td>
</tr>
<tr>
<td>24</td>
<td>KNN_Seed1</td>
<td>Nearest-Neighbour-Algorithmus mit k = 3 und Seedmenge nach Formel 17 und negativem Seetanteil von 5% bestimmt.</td>
<td>66</td>
</tr>
<tr>
<td>25</td>
<td>KNN_Seed2</td>
<td>Nearest-Neighbour-Algorithmus. Die Seedmengen wurden per Treshhold festgelegt. Oberer Schrankenwert liegt hierbei bei +45 unterer Schrankenwert bei -15.</td>
<td>66</td>
</tr>
<tr>
<td>26</td>
<td>CreateHistogramms</td>
<td>Erzeugung von 400 Histogrammen zu den Testsets.</td>
<td>67</td>
</tr>
</tbody>
</table>

Einsicht in den Quellcode der Programme kann beim Autor dieser Arbeit direkt beantragt werden.
F. Histogramm-Übersicht der Testdatensätze
<table>
<thead>
<tr>
<th>Größe der Teilmenge</th>
<th>5% Überlappung</th>
<th>25% Überlappung</th>
<th>50% Überlappung</th>
<th>75% Überlappung</th>
</tr>
</thead>
<tbody>
<tr>
<td>100:100</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1000:1000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10000:10000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>100000:100000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1000000:2000000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2000000:2000000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Gewicht: -40 -20 0 20 40 60
Danksagung


Doch nicht nur meinem persönlichen Umfeld gehört der Dank. Auch die Atmosphäre in der akademischen Umgebung war stets angenehm und ich kann über die Kollegen sowohl in der eigenen Arbeitsgruppe als auch des kompletten Institutes nur Gutes berichten.

Besonders möchte ich mich aber bei Herrn Stausberg bedanken, der mich für das Fach der Medizininformatik begeistern konnte, dem ich im Grunde genommen die Stelle als wissenschaftlicher Mitarbeiter zu verdanken habe und der mich stets mit vollstem Einsatz, was die Thematik dieser Arbeit anging, begleitet hatte. Die Jahre der gemeinsamen Zusammenarbeit sowohl als studentische Hilfskraft als auch als wissenschaftlicher Mitarbeiter werde ich in absolut positiver Erinnerung behalten.

Mit dieser Arbeit habe ich mir einen Traum erfüllt und ich möchte nochmals allen danken, die mir hierzu verholfen haben.
Ich erkläre hiermit an Eides statt,
dass ich die vorliegende Dissertation mit dem Thema

Einsatz und Optimierung einer überwachten Klassifizierungsmethode im Kontext eines
Privacy-Preserving-Record-Linkage

selbständig verfasst, mich außer der angegebenen keiner weiteren Hilfsmittel bedient und alle
Erkenntnisse, die aus dem Schrifttum ganz oder annähernd übernommen sind, als solche
kenntlich gemacht und nach ihrer Herkunft unter Bezeichnung der Fundstelle einzeln
nachgewiesen habe.

Ich erkläre des Weiteren, dass die hier vorgelegte Dissertation nicht in gleicher oder in
ähnlicher Form bei einer anderen Stelle zur Erlangung eines akademischen Grades eingereicht
wurde.

_________________________  ____________________________
Ort, Datum                     Unterschrift Doktorand